

PUB. NO.: 08-320796 [JP 8320796 A]  
PUBLISHED: December 03, 1996 (19961203)  
INVENTOR(s): YOSHIOKA SHINICHI  
KAWASAKI IKUYA  
MATSUI SHIGESUMI  
NARITA SUSUMU  
APPLICANT(s): HITACHI LTD [000510] (A Japanese Company or Corporation), JP  
(Japan)  
APPL. NO.: 07-240872 [JP 95240872]  
FILED: August 25, 1995 (19950825)  
INTL CLASS: [6] G06F-009/46; G06F-011/00; G06F-011/00  
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)  
JAPIO KEYWORD: R131 (INFORMATION PROCESSING -- Microcomputers &  
Microprocessors)

#### ABSTRACT

PURPOSE: To shorten the time of transition from exception generation to a handler which copes with it.

CONSTITUTION: The vector point of an exception processing handler regarding a TLB miss exception is obtained by adding a vector offset (H'400) to the vector base address of a register VBR. The vector point of an exception processing handler regarding an exception factor other than the TLB miss exception is obtained by adding the vector offset to the value (vector base address) of the register VBR. Then an exception factor code as an address offset which is obtained from a register EXPEVT or INTEVT is added to the obtained vector point. Consequently, branching to the necessary exception handler is performed to perform an exception process regarding the exception factor other than the TLB miss exception. The branch destination address need not be obtained by direct memory access.

特開平8-320796

(43) 公開日 平成8年(1996)12月3日

(51) Int.Cl. <sup>8</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/46	3 1 2		G 0 6 F 9/46	3 1 2
11/00	3 1 0		11/00	3 1 0 A
	3 3 0			3 3 0 D

審査請求 未請求 請求項の数11 F D (全 25 頁)

(21) 出願番号	特願平7-240872	(71) 出願人	000005108 株式会社日立製作所 東京都千代田区神田駿河台四丁目6番地
(22) 出願日	平成7年(1995)8月25日	(72) 発明者	吉岡 真一 東京都小平市上水本町5丁目20番1号 株式会社日立製作所半導体事業部内
(31) 優先権主張番号	特願平6-241991	(72) 発明者	川崎 郁也 東京都小平市上水本町5丁目20番1号 株式会社日立製作所半導体事業部内
(32) 優先日	平6(1994)9月9日	(72) 発明者	松井 重純 東京都小平市上水本町5丁目20番1号 株式会社日立製作所半導体事業部内
(33) 優先権主張国	日本 (J P)	(74) 代理人	弁理士 玉村 静世
(31) 優先権主張番号	特願平7-86067		
(32) 優先日	平7(1995)3月17日		
(33) 優先権主張国	日本 (J P)		

最終頁に続く

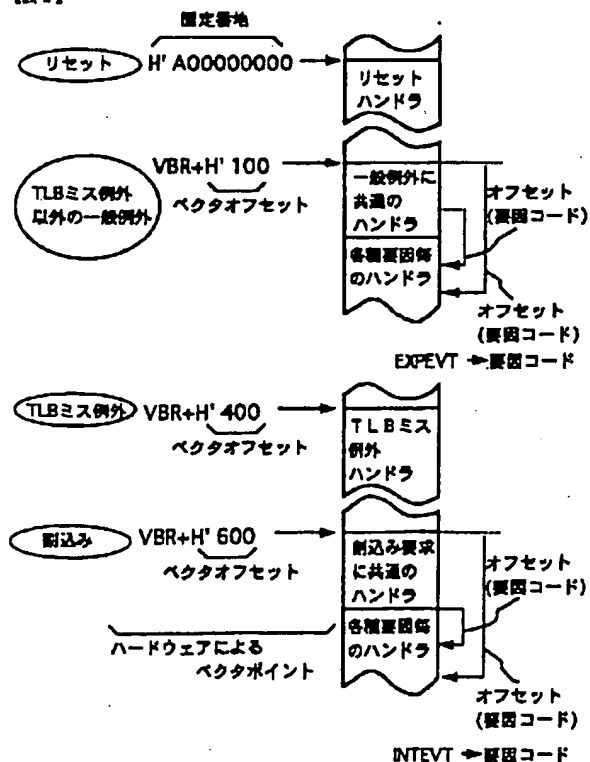
## (54) 【発明の名称】 データ処理装置

## (57) 【要約】

【目的】 例外発生からそれに対処するハンドラへの遷移時間を短縮する。

【構成】 T L B ミス例外に関する例外処理ハンドラのベクタポイントは、レジスタ V B R のベクタベースアドレスにベクタオフセット (H' 4 0 0) を加算し取得する。T L B ミス例外以外の例外要因に関する例外処理ハンドラのベクタポイントは、レジスタ V B R の値 (ベクタベースアドレス) にベクタオフセットを加算して取得される。そしてレジスタ EXPEVT 又は INTEVT から得たアドレスオフセットとしての例外要因コードが上記得られたベクタポイントに加算される。それにより、所要の例外ハンドラへ分岐し、T L B ミス例外以外の例外要因に関する例外処理が実行される。分岐先アドレスを直接メモリアクセスによって取得することを要しない。

【図2】



#### 【特許請求の範囲】

【請求項1】 例外事象の発生にตอบสนองし、上記例外事象に予め割り当てられた例外コードが書き込まれる記憶回路と、

プログラムカウンタと、上記例外事象の発生にตอบสนองし、所定の命令アドレスを上記プログラムカウンタに書き込み、上記命令アドレスに割り当てられた第1例外処理ハンドラを実行可能にする制御手段とを含む中央処理装置と、を含む、

上記制御手段は、上記第1例外処理ハンドラ内に規定される処理にしたがって、上記第1例外処理ハンドラから第2例外処理ハンドラへ分岐させるための第2命令アドレスを、前記記憶回路に書き込まれた上記要因コードをアドレスオフセットとして利用して算出させ、得られた第2命令アドレスを上記プログラムカウンタへ設定するものであることを特徴とするデータ処理装置。

【請求項2】 上記中央処理装置は、さらに、内部状態を格納するステータスレジスタと、

上記例外事象の発生にตอบสนองして、上記例外事象の発生時に上記ステータスレジスタが格納していたところの内部状態が書き込まれる第1退避レジスタと、

上記例外事象の発生にตอบสนองして、上記例外事象の発生時に上記プログラムカウンタが格納していて上記例外事象にตอบสนองする処理から復帰した時に実行されるべき命令を示している復帰命令アドレスが書き込まれる第2退避レジスタと、

を備えて成るものであることを特徴とする請求項1記載のデータ処理装置。

【請求項3】 上記中央処理装置は、さらに、算術論理演算回路と、上記例外事象の発生にตอบสนองして所定の値を発生する定数発生部と、外部メモリに記憶された複数の例外処理ハンドラのベースアドレスを格納するベースレジスタとを有し、

上記算術論理演算回路は、上記制御回路の制御の下で、上記ベースアドレスと上記所定の値とを加算し、上記所定の命令アドレスを発生するものであることを特徴とする請求項2記載のデータ処理装置。

【請求項4】 上記例外コードが書き込まれる記憶回路は、

上記中央処理装置の動作に同期して発生する例外事象に割り当てられた第1レジスタと、上記中央処理装置の動作と非同期で発生する例外事象に割り当てられた第2レジスタと、を含んで成るものであることを特徴とする請求項1乃至3の何れか1項記載のデータ処理装置。

【請求項5】 上記ステータスレジスタは、

上記データ処理装置の動作状態が、ユーザプログラムが走行されているユーザ状態か、あるいはシステムプログラムが走行されている特権状態かを択一的に示すデータが格納される第1制御ビットと、

上記例外事象の発生後に発生した他の例外事象を受け付

けるか無視するか否かを指示するデータが格納される第2制御ビットとを有し、

上記制御手段は、前記内部状態と上記復帰命令アドレスとを上記第1退避レジスタと上記第2退避レジスタへそれぞれ退避した後、上記第1制御ビットに特権状態を示すデータを設定し、上記第2制御ビットに上記例外事象の発生後に発生した他の例外事象を無視する事を示すデータを設定し、上記所定の命令アドレスに割り付けられた第1例外処理ハンドラに分岐させるものであることを特徴とする請求項3記載のデータ処理装置。

【請求項6】 上記制御手段は、上記第1例外処理ハンドラ内に規定される処理にしたがって上記第2制御ビットへデータを設定し、

上記第2制御ビットに設定されたデータが上記例外事象の発生後に発生した他の例外事象を受け付ける事を指示するデータの場合、上記制御手段は、上記第1及び第2退避レジスタの内容をメモリに退避させるものであることを特徴とする請求項5記載のデータ処理装置。

【請求項7】 さらに、複数のバンクを構成する第1及び第2汎用レジスタセットを有し、

上記第1及び第2汎用レジスタセットの一方は、前記特権状態における汎用レジスタとして利用され、上記第1及び第2汎用レジスタセットの他方は、前記ユーザ状態における汎用レジスタとして利用されて成るものであることを特徴とする請求項5記載のデータ処理装置。

【請求項8】 ブレークポイントを指定する命令アドレスが設定される命令ブレークアドレスレジスタを含み、上記命令ブレークアドレスレジスタに設定された命令アドレスに対応する命令が中央処理装置によって実行された時、命令ブレーク例外を発生する命令ブレークコントローラを有し、

上記第2制御ビットに上記例外事象の発生後に発生した他の例外事象を無視する事を示すデータが設定された状態において、上記制御手段が前記命令ブレーク例外を検出した時、上記制御手段は、前記第1及び第2退避レジスタへの内部状態及び復帰命令アドレスの退避を行うことなく、前記命令ブレーク例外に対応する命令ブレーク例外ハンドラに分岐させ、

上記命令ブレーク例外ハンドラは、上記内部状態及び上記復帰命令アドレスが退避されるべき上記第1及び第2退避レジスタの内容をメモリに退避した後、前記命令ブレークアドレスレジスタが保有するブレークポイントアドレスを利用して命令ブレーク例外ハンドラからの復帰アドレスを計算し、計算した復帰命令アドレスを前記第2退避レジスタに書き込み、これにより、多重例外を抑制した状態の例外ハンドラで、前記命令ブレーク例外を処理可能としたものであることを特徴とする請求項6記載のデータ処理装置。

【請求項9】 前記制御手段は、上記第2制御ビットに上記例外事象の発生後に発生した他の例外事象を無視す

る事を示すデータが設定された状態において、前記命令ブ레이크例外以外の第1の例外事象を検出したときは、前記要因レジスタにその要因コードを記録した後にリセット例外のハンドラに分岐させ、第2の例外事象を検出したときは前記他の例外事象を無視する状態が解除されるまで、第2の例外事象に応答する処理への分岐を抑止するものであることを特徴とする請求項8記載のデータ処理装置。

【請求項10】 例外事象の発生に応答することにより、中央処理装置の命令実行順序が変更され、その命令実行順序変更のために、ベクタベースアドレスに対する固定のオフセットであるベクタオフセットを、例外要因の数よりも少ない数だけ有するデータ処理装置であって、発生した例外事象の要因に対応する要因コードを記録する要因レジスタと、発生した例外事象の種別に応じて前記ベクタオフセットを生成する演算手段と、前記演算手段にて生成されたベクタオフセットとベクタベースアドレスによって規定される命令アドレスに処理を分岐させ、また、その分岐先において前記要因レジスタに記録されている要因コードを更に別の処理に分岐させるためのアドレスオフセットとして利用可能にする制御手段と、を含んで成るものであることを特徴とするデータ処理装置。

【請求項11】 アドレス変換バッファを備えたメモリマネジメントユニットを含み、前記演算手段で生成されるベクタオフセットは、複数の割り込み要因に割り当てられた第1のベクタオフセット、TLBミス例外要因に専用的に割り当てられた第2のベクタオフセット、TLBミス例外以外の一般例外に割り当てられた第3のベクタオフセットとされるものであることを特徴とする請求項10記載のデータ処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、リセット、例外事象や割り込み事象などの例外発生時に、その例外に対処するため例外処理プログラムを実行するデータ処理装置に関し、特に、例外の発生時点からその例外に対処するための例外処理ハンドラへ遷移するまでの遷移時間を短縮する技術に関する。本発明は、例えば、シングルチップマイクロコンピュータあるいはメモリ管理ユニットを内蔵するマイクロプロセッサに適用して有効な技術に関する。

【0002】

【従来の技術】上記データ処理装置に含まれる中央処理装置を用いたデータ処理の途中において、そのデータ処理装置の命令セット内に定義されていない未定義命令のデコード、無効な演算の発生、仮想記憶における記憶保護違反、TLBミスなどの一般的な例外事象（一般例外

事象とも言う：general exception events）が発生する場合や、データ処理装置の外部周辺回路におけるデータ入出力動作の終了を中央処理装置に通知するための終了通知やデータ処理装置内の通信モジュールからの受信要求などの割り込み要求（一般割り込み事象とも言う：general interrupt events）が発生する場合がある。

【0003】上記のような一般例外事象や一般割り込み事象などの例外（例外事象）が発生すると、上記中央処理装置は、データ処理プログラムの命令の実行を抑止し、発生した例外に対処するための例外処理ハンドラに制御を移行し、上記例外処理ハンドラによって規定されるデータ処理を実行し、発生した例外に対処する。上記例外処理ハンドラを実行した後、中央処理装置は、上記実行を抑止した命令を再実行し、あるいは、上記実行を抑止した命令の次の命令アドレスに復帰し、中断していた所定のデータ処理プログラムを継続する。そのため、一般例外事象や一般割り込み事象が発生した時、中央処理装置はその内部のプログラムカウンタの値やステータスレジスタの内部状態を外部メモリのスタック領域へ退避させるような動作を実行する。中央処理装置の処理が例外処理ハンドラの処理から上記抑止されたデータ処理プログラムへ復帰する時、中央処理装置は上記退避させたプログラムカウンタの値やステータスレジスタの内部状態を上記外部メモリのスタック領域からプログラムカウンタやステータスレジスタへそれぞれ格納し、上記抑止されたデータ処理プログラムを継続する。

【0004】所定のデータ処理プログラムから所定の例外処理ハンドラの処理への分岐手法としては、ハードウェア（論理回路）によって各種の分岐先アドレス（各種の例外処理ハンドラの前頭メモリアドレス）を固定する手法や、中央処理装置から分岐先アドレスを指定可能にするベクタ方式が採用されている。ベクタ方式は、例えば、割り込み要求に応答するための各種例外処理ハンドラの前頭アドレスを格納したベクタテーブルを外部メモリ上に配置し、中央処理装置から上記ベクタテーブルのポインタ（割り込みベクタレジスタ）を指定し、指定されたベクタテーブルから対応する例外処理ハンドラの前頭アドレスをリードし、リードした前頭アドレスの位置から所要の例外処理ハンドラをリードして実行する。

【0005】尚、割り込み要求などの例外について記載された文献の例としては、昭和62年12月25日に株式会社オーム社発行の「マイクロコンピュータハンドブック」第177頁及び第178頁がある。

【0006】

【発明が解決しようとする課題】しかしながら、上述のベクタ方式では、例外事象が発生してそれに対処するまでに、ベクタテーブルから対応する例外処理ハンドラの前頭アドレスを取得するための外部メモリのリード動作が必要となる。したがって、その外部メモリのリード動作の分だけ、例外事象の発生から対応ハンドラへ分岐す

るまでの遷移時間が長くなることがわかった。また、対応する例外処理ハンドラに分岐する前に、プログラムカウンタやステータスレジスタ、さらには汎用レジスタのデータなどを外部メモリ上のスタック領域に退避する操作を行う場合には、そのための外部メモリの書き込み動作によっても例外事象に対する応答が遅れることがわかった。

【0007】特に、中央処理装置の動作に同期して発生するTLBミスに関する例外に対して高速に応答することは、TLBミスの発生時点から上記抑止された命令の再実行までの時間を短縮できることを意味し、中央処理装置のデータ処理性能を向上させる上で究めて重要であることが本発明者によって見出された。なぜなら、TLBミスは、例外として分類されているものの、実際には、ユーザーの作成したデータ処理プログラムのプログラミングミスで発生するような本質的な例外と異なり、プログラミングミスの無いデータ処理プログラムの実行中において通常的に発生する事象である。したがって、TLBミスに高速に対処することが、中央処理装置のデータ処理性能を向上させる事になる。

【0008】更に、各種分岐先アドレスをハードウェアによって完全に固定にする手法も考えられるが、この手法はユーザ記述に係る例外処理ハンドラのマッピングやそのプログラムサイズに対する融通性がなく、使い勝手が悪いことがわかった。また、分岐先アドレスを生成するためのハードウェア量も増大すると考えられる。

【0009】本発明の目的は、例外事象の発生時点からそれに対処するための例外処理ハンドラへ遷移あるいは分岐するまでの遷移時間を短縮できるデータ処理装置を提供することにある。

【0010】本発明の別の目的は、例外事象に応答するための例外処理ハンドラの構成に自由度を持たせることができるデータ処理装置を提供することにある。

【0011】本発明の更に別の目的は、データ処理の高速化に密接なTLBミス例外のような処理に対しては、その処理への遷移時間を短縮することが可能であり、かつ、例外処理への遷移時間の短縮がTLBミス例外ほどデータ処理速度の高速化に寄与しないと考えられる例外事象や割り込み事象に対しては、例外処理ハンドラのメモリ上でのマッピングや例外処理ハンドラのメモリサイズに対する高い融通性を持たせることが可能なデータ処理装置を提供することにある。

【0012】本発明のその他の目的は、例外処理の点において物理的な回路規模の縮小とデータ処理の高速化との双方を満足させるた低コストかつ高速なデータ処理装置を提供することにある。

【0013】本発明の前記並びにその他の目的と新規な特徴は本明細書の記述及び添付図面から明らかになるであろう。

【0014】

【課題を解決するための手段】本願において開示される発明のうち代表的なものの概要を簡単に説明すれば下記の通りである。

【0015】データ処理装置は、図1に例示されるように、例外事象（例えばリセット、一般例外事象、一般割り込み要求）の発生に応答し、上記例外事象に予め割り当てられた例外コードが書き込まれる記憶回路（EXPEVT、INTEVT）と、プログラムカウンタ（PC）と、上記例外事象の発生に応答し、所定の命令アドレスを上記プログラムカウンタに書き込み、上記命令アドレスに割り当てられた第1例外処理ハンドラを実行可能にする制御手段（CTRL）とを含む中央処理装置（CPU）とを備える。

【0016】上記制御手段は、上記第1例外処理ハンドラ内に規定される処理にしたがって、上記第1例外処理ハンドラから第2例外処理ハンドラへ分岐させるための第2命令アドレスを、前記記憶回路に書き込まれた上記要因コードをアドレスオフセットとして利用して算出させ、得られた第2命令アドレスを上記プログラムカウンタへ設定する。

【0017】上記所定の命令アドレスへの分岐は、例えば、図2に例示されたハードウェアによるベクタポイントの操作とされる。上記所定の命令アドレス（分岐先のベクタポイント）に割り当てられる第1例外処理ハンドラは、例えば、図2に例示されるように、そのベクタポイントに固有の一つの一般例外事象であるTLBミス例外処理ハンドラ、或いはそのベクタポイントに固有の複数の例外事象（例えば、TLBミス例外以外の複数の一般例外事象）に共通の例外処理ハンドラ、又は複数の割り込みに共通の例外処理ハンドラとされる。上記共通の例外処理ハンドラにおいて、個々の例外に固有の別の例外処理ハンドラ（第2例外処理ハンドラ）へ分岐するためのアドレスオフセットとして、上記要因コードが利用される。上記要因コード（アドレスオフセット）に対するベースアドレスは、上記共通の例外処理ハンドラの記述によって決定される。ベースアドレスは、例えば、 $VBR$ （ベクタベースレジスタの値） $+H'100$ （記号 $H'$ は16進数であることを意味する）であっても、また、分岐時におけるプログラムカウンタの値をベースアドレスとし、さらには適当に演算して算出された値をベースアドレスにすることも可能である。

【0018】上記手段を別の観点より把握すれば、データ処理装置は、例外事象の発生に応じて中央処理装置による命令実行順序を変更するために、ベクタベースアドレスに対する固定のオフセットであるベクタオフセットを、例外要因の数よりも少ない数だけ有する。

【0019】本発明に従うデータ処理装置は、例外事象の発生時にステータスレジスタの内部状態が退避される第1退避レジスタ（SSR）と、プログラムカウンタ（PC）が保持していたところの例外処理から復帰した

後に実行されるべき命令を示している復帰命令アドレスが退避される第2退避レジスタ（SPC）と含む。上記第1及び第2退避レジスタの採用によって、ステータスレジスタの内部状態や復帰命令アドレスの退避のための外部メモリのメモリアクセス回数を減らすことができる。

【0020】さらに、本発明に従うデータ処理装置の中央処理装置は、図2に例示したハードウェアによるベクタポイントの操作のため、算術論理演算回路（ALU）と、上記例外の発生にตอบสนองして所定の値（ベクタオフセット）を発生する定数発生部あるいは演算手段（CVG、SFT）と、外部メモリに記憶された複数の例外処理ハンドラのベースアドレスを格納するベースレジスタ（VBR）とを有する。上記算術論理演算回路は、上記制御回路の制御の下で、上記ベースアドレスと上記所定の値とを加算し、上記所定の命令アドレス（分岐先命令アドレス）とを発生する。例えば、図2に従えば、TLBミス例外を除く一般例外のベクタオフセットはH'100とされ、TLBミス例外のベクタオフセットはH'400とされる。上記定数発生部は、2ビットの定数を発生する定数発生回路（CVG）と、上記定数発生回路から出力された2ビットの定数を所定の量だけシフトするシフト回路（SFT）とから構成され、その回路構成が簡単化されている。

【0021】要因コードを記録する記憶回路としてのレジスタは、中央処理装置（CPU）の動作に同期して発生する第1の例外事象（例えば一般例外）に割り当てられた第1レジスタ（EXPEVT）と、中央処理装置の動作と非同期で発生する第2の例外事象（例えば割り込み）に割り当てられた第2レジスタ（INTEVT）とされる。なお、リセットは、第2の例外事象の範疇に含まれる。

【0022】上記ステータスレジスタ（SR）は、上記データ処理装置の動作状態が、ユーザプログラムが走行されているユーザ状態か、あるいはシステムプログラムが走行されている特権状態かを択一的に示すデータが格納される第1制御ビット（MD）と、上記例外事象の発生後に発生した他の例外事象を受け付けるか無視する（マスク）か否かを指示するデータが格納される第2制御ビット（BL）とを有する。

【0023】上記制御手段は、前記内部状態と上記復帰命令アドレスとを上記第1退避レジスタ（SSR）と上記第2退避レジスタ（SPC）へそれぞれ退避した後、上記第1制御ビットに特権状態を示すデータを設定し、上記第2制御ビットに上記例外事象の発生後に発生した他の例外事象を無視する事示すデータを設定し、上記所定の命令アドレスに割り付けられた第1例外処理ハンドラに分岐させる。上記第1及び第2制御ビットに対するアクセスは、所定の特権命令（例えばLDC、STC）にて可能にされる。即ち、図2に例示されように、

ベクタポイントに割り当てられた上記第1例外処理ハンドラ上でのプロセッサモードは、特権状態及び例外の多重受け付け抑止の状態とされる。ここで、特権状態は、ユーザ状態ではアドレスエラーとされるアドレス空間をアクセス可能な状態であり、ユーザ状態では実行不可能な特権命令（例えばLDC、STC）を実行することができる、という点においてユーザ状態と相違される。

【0024】例外発生時における内部状態の退避と復帰アドレスの退避を第1及び第2退避レジスタにできるようにするため、ベクタポイントに割り当てられた上記第1例外処理ハンドラ上では、新たに発生する例外の多重受け付けを抑止するか否かを決定してその内容を上記制御ビットに反映し、多重に受け付けるときは上記第1及び第2の退避レジスタの内容をメモリに退避させる。

【0025】例外発生時の汎用レジスタのメモリ退避を低減させ、或は、汎用レジスタに対する処理の自由度を向上させるため、複数のバンクを構成する第1及び第2汎用レジスタセットが設けられる。上記第1及び第2汎用レジスタセットの一方は、前記特権状態における汎用レジスタとして利用され、上記第1及び第2汎用レジスタセットの他方は、前記ユーザ状態における汎用レジスタとして利用される。汎用レジスタのレジスタバンクを前記特権状態においてのみソフトウェアで切り換え可能にするといふ。

【0026】本発明に従うデータ処理装置は、さらに、命令ブレークコントローラ（UBC）を有する。命令ブレークコントローラ（UBC）は、ブレークポイントアドレスが設定される命令ブレークアドレスレジスタ（IBR）を含み、上記命令ブレークアドレスレジスタに設定された命令アドレスに対応する命令が中央処理装置によって実行された時、命令ブレーク例外を発生する。前記制御手段は、例外の多重受け付け抑止が指示された状態において前記命令ブレーク例外を検出したとき、前記第1及び第2退避レジスタへの内部状態及び復帰命令アドレスの退避を行わずに命令ブレーク例外ハンドラに分岐させる。、上記命令ブレーク例外ハンドラは、上記状態及び復帰アドレス退避用の退避レジスタの内容をメモリに退避した後、前記命令ブレークアドレスレジスタが保有するブレークポイントアドレスを利用して命令ブレーク例外ハンドラからの復帰命令アドレスを計算し、計算した復帰命令アドレスを前記第2退避レジスタに書き込む。これにより、前記例外の多重受け付けが抑止された状態の例外処理ハンドラで、前記命令ブレーク例外を処理可能とする。

【0027】上記において、前記制御手段は、更に、例外の多重受け付けを抑止した状態において、前記命令ブレーク例外以外の第1の例外を検出したときは、前記要因レジスタにその要因コードを記録した後にリセット例外のハンドラに分岐させ、また、第2の例外を検出したときは、例外の多重受け付け抑止の状態が解除されるま

で第2の例外の受け付けを抑止する。

【0028】

【作用】上記した手段によれば、ベクタポイント割り付けられたら第1例外処理ハンドラは、ある例外要因（例えばTLBミス例外）に対してはそれ固有の一つの例外ハンドラとされ、他の例外要因（例えばTLBミス例外以外の一般例外）に対しては夫々の要因に対応して設けられた別の例外処理ハンドラとされる第2例外処理ハンドラ（例えば、アドレスエラー例外、TLBプロテクト違反例外）へ分岐させるための記述を含む共通の例外処理ハンドラとされる。

【0029】前者においては、メモリアクセスを伴わずにハードウェアによる処理だけで特定の第1例外処理ハンドラに分岐できるので、所望の例外処理へ高速に移行することが可能になる。後者においては、さらに別のハンドラに分岐するためのアドレスオフセットとして記憶回路（要因レジスタ）の例外要因コードを利用するので、分岐先アドレスの取得にメモリ上のアドレステーブルをアクセスすることが必要とされない。従って、この場合にも、所望の例外処理ハンドラへの高速な移行が可能なる。

【0030】見方を変えれば、後者において要因レジスタに保持されている例外要因コードは、さらに別の例外処理ハンドラに分岐するためのアドレスオフセットとして利用できるようにコード割り当てが行われているということである。例えば、アドレス換算で32バイト分に相当するようなH'20の間隔を置いてコード割り当てが行われている。

【0031】更に、後者において、例外要因コードは分岐のためのアドレスオフセットとして利用されるので、当該オフセットに対するベースアドレスはハードウェアによって決定されているベクタポイントに割り当てられた第1例外処理ハンドラの記述に従って自由に決定でき、実際の分岐先アドレスや分岐先の例外処理ハンドラのサイズに対して自由度を保証することができる。

【0032】内部状態及び復帰命令アドレスを退避するレジスタ（SSR、SPC）を採用することは、例外発生時の退避に際してメモリアクセスを減らす。

【0033】中央処理装置の動作クロックの周波数が割り込み要求の発生元とされる周辺モジュールの動作クロックの周波数に対して複数倍のクロック周波数で動作される場合、前記中央処理装置の動作に同期して発生される第1の例外事象としての一般例外と非同期で発生される第2の例外事象としての割り込み要求との双方に対処するため、例外要因コードを格納するレジスタは上記第1例外事象と上記第2例外事象に対して別々に設ける。双方の例外発生に対して同じようなタイミングで例外要因コードを共通のレジスタに書き込むための煩雑な処理を避けられるようにする。

【0034】例外要因に応じてハードウェアで決定され

ている分岐先のハンドラ（第1例外処理ハンドラ）上での初期的なプロセッサモードを、例外事象の多重受け付け抑止状態及び特権状態によって一定にすることは、ユーザが当該ハンドラ上で規定できる例外処理の内容に高い自由度を保証できる。

【0035】汎用レジスタのレジスタバンクを特権状態においてのみソフトウェアで切り換え可能にすることは、特権状態においてユーザ状態とは異なるバンクの汎用レジスタを利用できるようになり、例えば、例外処理においてユーザ状態から特権状態にプロセッサモードを切り換えるときに汎用レジスタの内容をメモリに退避しなくても済むようにでき、例外処理への移行を高速化できる。

【0036】例外を抑止した状態の例外ハンドラ中で命令ブレーク例外を処理可能にすることは、多重に発生する例外の受け付けが禁止されている状態のデバッグ対象プログラムに対しても任意の位置で命令ブレークをかけてシステム評価やプログラムデバッグを行うことを保証できる。

【0037】

【実施例】

《マイクロコンピュータの概要》 図1は、本発明の実施例に係るシングルチップマイクロコンピュータ（シングルチップマイクロプロセッサとも言う）のブロック図を示す。同図に示されるマイクロコンピュータMPUは、例えば、公知の半導体集積回路製造技術によって単結晶シリコンのような1個の半導体基板あるいは半導体チップに形成される。マイクロコンピュータMPUは、特に制限されないが、内部バスとしてシステムバスS-bus、キャッシュバスC-busおよびペリフェラルバスP-busを含む。すなわち、マイクロコンピュータMPUは、スリーバス構成を有する。内部バスS-bus、C-bus、P-busの各々は、データを転送するためのデータバス、アドレス信号を転送するためのアドレスバス及び制御信号を転送するための制御バスを備えている。

【0038】システムバスS-busには、中央処理装置（central processing unit）CPU、乗算器（multiplier）MLT、キャッシュメモリ（cache memory）CACHE、メモリアネージメントユニット（memory management unit）MMU、及び命令ブレークコントローラ（instruction break controller）UBCが結合される。キャッシュバスC-busには上記キャッシュメモリCACHE、メモリアネージメントユニットMMU、命令ブレークコントローラUBC、及びバスステートコントローラ（bus state controller）BSCが結合される。バスステートコントローラBSCに接続されるペリフェラルバスP-busには、タイマTMU、中央処理装置CPUへの動作クロックの供給が停止されても計時動作が可能にされるリアルタイムクロック回路（real time clock cir

cuit) R T C、外部メモリ M M R Yに含まれるダイナミック型メモリのリフレッシュ動作を制御するリフレッシュコントローラ(refresh controller) R E F C、及びマイクロコンピュータ M P Uの外部に設けられた外部周辺装置(external peripheral devices)との間のシリアル通信を実行するシリアルコミュニケーションインタフェース(serial communication interface) S C Iなどの内蔵周辺モジュール(internal peripheral modules)が接続される。また、上記バスステートコントローラ B S Cは、入出力回路(input and output circuit) E X I Fを介して、外部バス E X - b u sに接続可能にされる。上記外部バス E X - b u sには、外部メモリ M M R Yや補助記憶装置 D I S Kなどが接続されることになる。上記バスステートコントローラ B S Cは、上記内蔵周辺モジュール及び外部に対するバスサイクルの起動とそれに伴う各種バス制御を行う。

【0039】割り込みコントローラ(interrupt controller) I N T Cは、上記内蔵周辺モジュールから、及び複数ビットの外部割り込み端子(external interrupt terminals) I R L 0 ~ I R L 3から外部割り込み要求(external interrupt request)を受け、割り込み優先レベル(interrupt priority level)に従って割り込み要求の調停などを行う。割り込みコントローラ I N T Cは、発生した割り込み要求の受け付けを許可する場合、上記中央処理装置 C P Uへ割り込み要求信号 S I G 1を供給する。また、上記割り込みコントローラ I N T Cは、受け付けられた割り込み要求の割り込み要因を割り込み要因信号 S I G 2によってメモリマネージメントユニット M M Uの制御回路(T L B C)へ通知する。上記割り込み要求信号 S I G 1を受けた中央処理装置 C P Uは、割り込み許可信号 S I G 3をメモリマネージメントユニット M M Uの制御回路 T L B Cへ供給する。上記メモリマネージメントユニット M M Uの制御回路 T L B Cは、割り込み許可信号 S I G 3に応答し、許可された割り込み要求に対応する例外要因コード(exception code)を後述する記憶回路としての割り込み要因レジスタ I N T E V Tに書き込む。中央処理装置 C P Uの制御部 C T R Lは、上記割り込み要因レジスタ I N T E V Tにセットされた例外要因コードを用いて、実行中のデータ処理プログラムの処理から所定の割り込み処理へその処理を分岐する。

【0040】本実施例のマイクロコンピュータ M P Uは、論理アドレス空間(logical address space)を論理ページ(logical page)と呼ばれる単位に分割し、そのページ単位に物理アドレス(physical address)へのアドレス変換(address translation)を行うための仮想記憶(virtual memory)をサポートする。上記メモリマネージメントユニット M M Uのアドレス変換バッファ(address translation look-aside buffer) T L Bは、論理ページ番号と物理ページ番号とに関する複数の

変換対を T L B エントリとして格納する。上記メモリマネージメントユニット M M Uの制御部 T L B Cは、中央処理装置 C P Uが出力する論理アドレスをアドレス変換バッファ T L Bを用いて物理アドレスに変換するための制御を行う。T L B ミスの場合、すなわち、中央処理装置 C P Uが出力した論理アドレスに対応する変換対が T L B エントリとしてアドレス変換バッファ T L B 内に記憶されていなかった場合、制御部 T L B Cは、外部メモリ M M R Y上のページテーブルをアクセスしてその論理アドレスに対応する変換対を外部メモリ M M R Yから読み込むように動作する。その後、制御部 T L B Cは、読み出された変換対を T L B エントリとしてアドレス変換バッファ T L Bへ書き込むように動作する。

【0041】アドレス変換バッファ T L Bは、例えば、4ウェイ・セットアソシアティブ(4way-set-associative)方式のキャッシュメモリ(cache memory)によって構成される。T L B ミスなどのアドレス変換に係る後述されるような各種例外事象が発生すると、制御部 T L B Cはその例外事象に対応する例外要因コードを後述する記憶回路としての例外要因レジスタ E X P E V Tにセットし、且つ、T L B ミスなどのアドレス変換に係る例外事象の発生を通知するための通知信号 S I G 4を中央処理装置 C P Uの制御部 C T R Lへ供給する。中央処理装置 C P Uは、例外要因レジスタ E X P E V Tにセットされた例外要因コードを用いて、或いは、例外要因レジスタ E X P E V Tにセットされた例外要因コードを用いずにハードウェアで直接的に、実行中のデータ処理プログラムの処理から対応する例外処理へその処理を分岐する。

【0042】中央処理装置 C P Uは、例えば、4ギガバイトの論理アドレス空間をサポートするために、32ビットのアドレス信号を利用する。図1に示されるように、中央処理装置 C P Uは、例えば、汎用レジスタ(general purpose registers) R 0 ~ R 1 5や算術論理演算器(arithmetic and logic operation unit) A L Uを含む演算部(operation unit) 1 0 0と、後述されるプログラムカウンタ(program counter) P Cなどのシステム制御用レジスタセット(system control register set) 1 1 0と、命令のフェッチ、命令の解読、命令の実行手順の制御および演算の制御を行う制御部 C T R Lを含む。これらの回路装置(R 0 ~ R 1 5、A L U、1 1 0、C T R L)は、システムバス S - b u sに結合される。中央処理装置 C P Uは、外部メモリ M M R Yから実行されるべき命令をフェッチし、その命令記述(instruction code)に応じたデータ処理を行う。図1において、制御信号 S I G 5は、中央処理装置 C P Uからメモリマネージメントユニット M M Uへの各種制御信号及び中央処理装置 C P Uの内部状態を通知するための信号を総称するものである。

【0043】キャッシュメモリ C A C H Eは、特に制限



されないが、4ウェイ・セットアソシアティブ形式とされる。キャッシュメモリCACHEに対するインデックスは、中央処理装置CPUから出力された論理アドレスの一部を用いて行われ、キャッシュメモリCACHEのキャッシュエントリのタグ部には物理アドレスが保有される。インデックスされたタグ部から出力される論理アドレスはアドレス変換バッファTLBで変換された物理アドレスと比較され、その比較結果に応じてキャッシュミス/ヒットが判定される。キャッシュミスの場合、当該キャッシュミスに係るデータ又は命令は外部メモリから読み込まれ、読み込まれたデータ又は命令は新たなキャッシュエントリとしてキャッシュメモリCACHEに格納される。

【0044】上記命令ブレイクコントローラUBCは、デバッグ機能を強化するために設けられており、システムバスS-busの状態がブレイク条件に一致するかを監視する。システムバスS-busの状態がブレイク条件と一致する場合、中央処理装置CPUにブレイク割込みを発生させる。命令ブレイクコントローラUBCに含まれるIBRは、ブレイク条件として命令アドレスなどが設定される命令ブレイクアドレスレジスタである。中央処理装置CPUは、デバッグ若しくはエミュレーション開始前にそのためのサービスルーチンを実行し、命令ブレイクアドレスレジスタIBRにブレイクすべき命令の先頭アドレスやオペランドアドレスなどの所望の命令ブレイク条件をあらかじめ設定する。マイクロコンピュータMPUの内部状態が前記命令ブレイク条件に一致すると、後述の命令ブレイク例外が発生する。これによって、中央処理装置CPUは、デバッグのための命令ブレイクハンドラを実行可能にされる。したがって、マイクロコンピュータMPUの内部でブレイクポイント制御を行うことができる。

【0045】《CPUのレジスタ構成》次に、中央処理装置CPUにおけるレジスタ構成が説明される。汎用レジスタR0～R15は、図3に示されるように、夫々32ビットの記憶ビットを有し、図示されないインターフェイス回路を介して、上記システムバスに結合される。汎用レジスタR0～R7は、プロセッサモードによって切り換えられるバンクレジスタとされる。換言すれば、汎用レジスタR0～R7はバンク0とバンク1の2セット備えられる。すなわち、第1汎用レジスタセットとしてのR0 (BANK0)～R7 (BANK0)と第2汎用レジスタセットとしてのR0 (BANK1)～R7 (BANK1)とされる。上記プロセッサモードは、ユーザのアプリケーションプログラムが走行される動作状態を意味するユーザモード (ユーザ状態) と、オペレーティングシステム (operating system) などのシステムプログラムが走行される動作状態を意味する特権モード (特権状態) とを含む。したがって、特権状態とユーザ状態の夫々は、固有の汎用レジスタR0～R7を有す

ることができる。即ち、図4に示されるユーザ状態においては、バンク0の汎用レジスタR0 (BANK0)～R7 (BANK0)が利用可能にされる。特権状態においては、後述されるステータスレジスタSRのレジスタバンクビットRB (SR, RB)の設定状態によって汎用レジスタR0～R7の利用形態が決定される。

【0046】例えば、図5に示されるように、特権状態において、RB=0のときは、R0 (BANK0)～R7 (BANK0)は汎用レジスタとして自由な利用が許容され、R0 (BANK1)～R7 (BANK1)に対しては制御ロード命令 (LDC) 及び制御ストア命令 (STC) によるアクセスのみが許容される。RB=1ときは、上記とは逆にR0 (BANK1)～R7 (BANK1)が汎用レジスタとしての自由な利用が許容され、R0 (BANK0)～R7 (BANK0)に対しては制御ロード命令 (LDC) 及び制御ストア命令 (STC) によるアクセスのみが許容される。前記制御ロード命令 (LDC) 及び制御ストア命令 (STC) は、特権状態において実行可能な特権命令若しくはシステム制御命令の一例である。

【0047】このように汎用レジスタをバンクレジスタ構成とし、特権状態においてユーザ状態とは異なるバンクの汎用レジスタを利用可能にすることは、例えば、例外処理においてユーザ状態から特権状態にプロセッサモードを切り換える時、ユーザ状態において汎用的に使用されていた汎用レジスタR0～R7の内容を外部メモリMMRYのスタック領域へ退避しなくても済むようにできる。したがって、中央処理装置CPUの例外処理への移行が高速化できる。

【0048】システムレジスタは、図3に示されるように、データレジスタハイMACH、データレジスタローMACL、復帰アドレスレジスタPR、及びプログラムカウンタPCとされる。データレジスタハイMACH及びデータレジスタローMACLは、乗算、積算や積和演算のためのデータを格納する。復帰アドレスレジスタPRは、サブルーチンからの戻りアドレスを格納する。プログラムカウンタPCは、現在の命令のスタート番地を指す。

【0049】コントロールレジスタは、図6に示されるように、セーブステータスレジスタSSR、セーブプログラムカウンタSPC、グローバルベースレジスタGBR、ベクタベースレジスタVBR、及びステータスレジスタSRとされる。セーブステータスレジスタSSRは、例外事象の発生時などにおいて、現在のステータスレジスタSRの値を退避するレジスタである。セーブプログラムカウンタSPCは、例外事象が発生したときに、対応する例外処理から復帰した後に中央処理装置CPUが実行すべき命令の命令アドレスを保持するためのレジスタである。セーブプログラムカウンタSPCには、例外事象が発生したとき所定のタイミングでプログ

ラムカウンタPCの値が退避される。グローバルベースレジスタGBRは、GBR-インダイレクトアドレスモードにおけるベースアドレスを格納するレジスタである。GBR-インダイレクトアドレスモードは、シリアルコミュニケーションインタフェースSCIなどの上記内蔵周辺モジュールのレジスタ領域にデータを転送する時、また、論理演算の時に利用される。ベクタベースレジスタVBRは、例外処理のためのベクタ領域のベースアドレス（ベクタテーブルベースアドレス）を保有する。ステータスレジスタSRは、演算においてキャリー、ボロー、オーバーフローを示すために利用されるTビット、メモリアクセス制御に利用されるSビット、割り込み要求に対するマスクレベルを4ビットで示す割り込みマスクビットIMASK、除算に利用されるM及びQビット、上述のレジスタバンクビットRB、ブロックビットBL、プロセッサオペレーションモードビットMD、及びゼロビットを含む。ブロックビットBLは例外をマスクするために利用され、BL=1は例外をマスクすることを指示し、BL=0は例外を許容することを指示する。モードビットMDはMD=1にて特権モードを、MD=0にてユーザモードを示す。上記M、Q、S、Tの各ビットは、ユーザモードにおいて、所定の専用命令を実行した中央処理装置CPUによってセットあるいはクリアすることが可能である。その他全てのビットは、特権モードにおいてだけ中央処理装置CPUによってリード・ライト可能にされる。上記コントロールレジスタに対する書き込みは制御ロード命令(LDC)を実行する中央処理装置CPUによって行われ、上記コントロールレジスタに対する読み出しは制御ストア命令(STC)を実行する中央処理装置CPUによって行われる。

#### 【0050】《マイクロコンピュータのアドレス空間》

本実施例のマイクロコンピュータMPUは、4GB（ギガバイト）の論理アドレス空間をサポートするために32ビットのアドレスを利用する。該論理アドレスは、空間番号によって拡張可能にされる。図13は、特権モードのアドレス空間とユーザモードのアドレス空間とのアドレスマッピングを示している。同図において、“Mapped”と示される領域は、アドレス変換バッファTLBを利用したアドレス変換の対象とされる。

【0051】H'FFFFFFFF~H'80000000の領域は、特権モードにおいて、アクセス可能な領域とされ、ユーザモードでのアクセスはアドレスエラーとされる。

【0052】アドレス領域P4はコントロールスペースであり、周辺モジュールなどのコントロールレジスタなどがマッピングされる。

【0053】アドレス領域P1、P2は、固定された物理アドレスを有する領域であり、アドレス変換バッファTLBを利用したアドレス変換の対象にはされない。ア

ドレス領域P1、P2の論理アドレスは、一定の定数の加算又は減算によって物理アドレスに変換される。したがって、特権状態において、アドレス領域P1、P2をアクセスするとき、TLBミスを初めとするアドレス変換に係る例外事象が発生することはない。一般割り込み事象や一般例外事象などの各種例外ハンドラの物理アドレスをアドレス領域P1又はP2に割り当てることにより、TLBミスを初めとするアドレス変換に係る例外事象が例外処理中に新たに発生することがなくなる。換言すれば、アドレス変換に係る例外事象が、多重に発生する事態を回避でき、これによってデータ処理の効率化に寄与することができる。

【0054】特に、アドレス領域P2はキャッシュメモリCACHEによるキャッシュの対象とはされず、一方、アドレス領域P1はキャッシュの対象とされる。例外処理ハンドラに関して言えば、複数存在する例外処理ハンドラの内どの例外処理ハンドラをアドレス領域P1又はP2へ割り当てるかは、その例外処理の性質に応じて決定される。例えば、TLBミス例外に対処するための例外処理ハンドラのように高速処理が要求される例外処理ハンドラは、アドレス領域P1に割り当てるのが得策である。一方、例外処理の高速性が要求されない例外処理ハンドラは、キャッシュメモリを別のデータに解放しておくために、アドレス領域P2に割り当てるのが得策である。

【0055】《例外処理の概要》 例外処理は、特別なハンドリングが要求されるので、現在のプログラム実行を例外的に別の処理へ分岐させる処理であり、中央処理装置CPUは現在の命令（その実行は後述の割り込み要求に対しては完了まで継続することが許容される）実行を抑止し且つユーザ記述による例外ハンドラにその制御を移行することによってその例外処理の要求に応答する。例えば、例外処理は、現在実行中の命令との関係では、3つの処理に分類できる。第1の処理は、現在実行中の命令を無視して、例外ハンドラの処理へ分岐する処理である。第2の処理は、現在実行中の命令を、例外ハンドラの処理完了後に、再実行する処理である。第3の処理は、現在実行中の命令実行を完了した後に、例外ハンドラの処理へ分岐して分岐先での処理完了後に分岐前の状態に復帰する処理である。後者の二つについては、一般例外のような第1の例外事象と所謂割り込み要求のような第2の例外事象として区別することができる。割り込み要求のような第2の例外事象は、中央処理装置CPUの動作とは非同期で発生するので、実行中の命令が実行完了された段階で受け付けられ、割り込み処理に対応する例外処理ハンドラの処理が終了すると、当該実行完了された命令の次の命令に中央処理装置CPUの処理が復帰される。一般例外のような第1の例外事象は、中央処理装置CPUの動作に同期して発生するので、その要求の原因が取り除かれた後、当該例外処理の要求が発

せられた時点で中央処理装置CPUの実行していた命令が中央処理装置CPUによって再実行される。ただし、全ての例外処理には双方の定義に合致しないものが存在することは言うまでもない。

【0056】図7は、ベクタとして割り付けられた例外事象を示している。例外処理は、図7に示されるように、リセット処理、中央処理装置CPUの動作に同期して発生する例外事象（一般例外）に対する処理、中央処理装置CPUの動作と非同期で発生する割り込み要求に対する処理に大別される。夫々の例外処理において、現在実行中の命令に対する処置は、同図に示されるように、無視 (Aborted)、再実行 (Retried) および完了 (Completed) の3種類に大別される。

【0057】1) リセット処理は、パワーオンリセット (Power-On) とマニュアルリセット (Manual Reset) を含む。例えば、パワーオンリセットは、電源投入によってリセット信号がアサートされることを条件に、中央処理装置CPUや内蔵周辺モジュールの初期化を行うリセット処理とされる。

【0058】2) 一般例外は、中央処理装置CPUの内部で発生する例外と、メモリマネージメントユニットMMUによるアドレス変換に際して発生される例外に大別される。

【0059】前者の例外事象として、予約命令例外 (Reserved Instruction Exception)、スロット不当命令例外 (Illegal Slot Instruction Exception)、トラップ命令例外 (Unconditional Trap)、及び命令ブレイク例外 (User Breakpoint Trap) がある。

【0060】後者の例外事象として、命令アクセス時とデータアクセス時におけるアドレスエラー例外 (Address Error)、命令アクセス時とデータアクセス時におけるTLBミス例外 (TLB Miss)、命令アクセス時とデータアクセス時におけるTLBインバリッド例外 (TLB Invalid)、命令アクセス時とデータアクセス時におけるTLBプロテクト違反例外 (TLB Protection Violation)、TLBイニシャルページライト例外 (Initial Page Write) がある。

【0061】上記TLBミスは、アドレス変換バッファTLBのアドレス比較が不一致である場合に発生し、当該不一致に係る論理アドレスのページテーブルエントリを外部メモリMMRYからアドレス変換バッファTLBにロードする処理とされる。トラップ命令例外は、TRAP命令を実行することを条件に発生し、当該命令のイミディエイト値を用いた分岐先のハンドラによって規定される処理を行うものであり、ソフトウェア割り込みとして位置づけることができる。予約命令例外は、遅延分岐命令を利用するときに、遅延スロット以外にある未定義命令コードをデコードすることなどを条件に発生し、分岐先のハンドラで規定される処理を行う。命令ブレイク例外は、レジスタIBRに設定されたブレイク条件が

一致したことを条件に発生し、分岐先のハンドラで規定される処理を行う。

【0062】3) 割り込み要求は、マスク不可能な割り込み (Nonmaskable Interrupt)、外部割り込み (External Hardware Interrupt)、及び内蔵周辺モジュール割り込み (Peripheral Module Interrupt) を含む。

【0063】例外処理の優先レベル (Priority Level) に関しては、特に制限されないが、図7に示されるように、以下のように定義される。すなわち、リセットは優先レベル1に割り当てられ、一般例外は優先レベル3に割り当てられ、マスク不可能な割り込み要求は優先度レベル2に割り当てられ、その他の割り込み要求は優先度レベル4に割り当てられる。この優先レベルは、高い順に1~4とされ、最も高い優先レベルは1とされ、最も低い優先レベルは4とされる。

【0064】上記一般例外は、命令の実行シーケンスに関係して発生するので、優先レベル3が割り当てられている。優先レベル3に割り当てられている各種一般例外には、発生する命令実行段階の早遅に従って、1~12の実行優先度 (Execution Order) が割り当てられる。命令がパイプライン的に実行される場合、ある命令の実行中に検出された優先レベル3のある一般例外は、その後の命令の実行中に検出された優先レベル3の他の一般例外に先行して受け付けられるように制御される。従って、実行優先度は、個々の命令の実行スパン内での優先度を規定しているとみなされる。特に、命令ブレイク例外 (ユーザブレイクトラップ例外) において、ブレイクポイントとして指定されている命令の実行前にトラップを発生させる場合、命令ブレイク例外の実行優先度は1とされ、実行後にトラップを発生させる場合、命令ブレイク例外の実行優先度は12、オペランドブレイクポイントが設定されている場合も命令ブレイク例外の実行優先度は12とされる。外部割り込みと内蔵周辺モジュール割り込みとの間での優先レベルはソフトウェアによって定義される。

【0065】尚、図7における一般例外内のある1部の一般例外は、命令アクセス (Instr. Access) 時とデータアクセス (Data Access) 時とで区別されるように図示されている。これは、図7のAddress ErrorからInitial Page Writeまでの一般例外の記載は、ある1つの命令をパイプライン的に実行するときの実行順序に従って例外要因を記載してあるためである。したがって、図7に示されるTLB Miss (Instr. Access) とTLB Miss (Data Access) は相互に同一の例外ハンドラで処理される。Address Error (Instr. Access) とAddress Error (Data Access)、TLBInvalid (Instr. Access) とTLB Invalid (Data Access) そしてTLB Protection Violation (Instr. Access) とTLB Protection Violation (Data Access) も同様である。また、図7においては、一般例外は優先レベル3に割り当てられ、マスク不可能な割り込み

要求は優先度レベル2に割り当てられているが、一般例外に優先レベル2を割り当て、マスク不可能な割り込み要求に優先度レベル3を割り当てても良い。

【0066】《例外処理のためのベクタ配置》 夫々の例外処理の内容を規定する各種例外処理ハンドラ（例外ハンドラと総称する）は、ベクタ配置を利用して呼び出される。マイクロコンピュータMPUは、そのハードウェア構成によって4種類のベクタ配置をサポートする。固定された物理アドレスH' A 0 0 0 0 0 0 0 はリセットのためのベクタとして割り当てられる。他の例外処理に対しては、ソフトウェアによって上記ベクタベースレジスタVBRに設定されたベクタテーブルベースアドレスからの固定番地のオフセット（ベクタオフセット）が割り当てられている。それぞれのベクタオフセットは、図7に示されるように、H' 0 0 0 0 0 1 0 0、H' 0 0 0 0 0 4 0 0、H' 0 0 0 0 0 6 0 0とされる。上記TLBミス例外のベクタオフセットにはH' 0 0 0 0 0 4 0 0が割り当てられ、TLBミス例外以外の一般例外のベクタオフセットにはH' 0 0 0 0 0 1 0 0が割り当てられ、割り込み要求のベクタオフセットにはH' 0 0 0 0 0 6 0 0が割り当てられる。

【0067】ベクタテーブルのベースアドレスは、ベクタベースレジスタVBRにソフトウェアを実行する中央処理装置CPUによってロードされる。例えば、タスク切換え若しくはプロセス切換えに際して、オペレーティングシステムのようなシステムプログラムを実行する中央処理装置CPUによってベクタベースレジスタVBRの設定が行われる。ベクタテーブルのベースアドレスは、所定の固定された物理アドレス空間に配置されることになる。即ち、ベクタテーブルのベースアドレスは前記アドレス領域P1、P2に配置される。上記レジスタVBRの値とベクタオフセットとによって決定されるベクタポイント（ベクタアドレス）は、ハードウェアによって規定される。即ち、ベクタポイントは、ユーザ記述によるプログラムによって規定される性質のものではない。

【0068】図14は、前記固定オフセットとしてのベクタオフセットの値を生成するためのハードウェアの構成を示している。同図において、定数発生回路CVGとシフトSFTは、ベクタオフセットを生成するために設けられている定数発生部或は演算手段みなされる。ベクタオフセットの値とベクタベースレジスタVBRの値との加算は、算術論理演算器ALUによって行われ、加算して得られたベクタポイントは、プログラムカウンタPCに書き込まれる。その後、中央処理装置CPUは、プログラムカウンタPCに書き込まれたベクタポイントに割り当てられる例外処理ハンドラをアクセスするため、システムバスS-bus内のアドレスバスへ上記ベクタポイント（論理アドレス）を発生する。メモリ管理ユニットMMUは、上記ベクタポイントとされる論理ア

ドレスを受け、その論理アドレスに関するアドレス変換対をエントリとして記憶している場合、上記論理アドレスを物理アドレスに変換し、キャッシュメモリCACHEへ供給する。キャッシュメモリCACHEに上記物理アドレスに対応する命令が存在しない場合、上記物理アドレスが、キャッシュバスC-busのアドレスバス、バスステートコントローラBSC、入出力回路EXIF及び外部バスEX-busのアドレスバスを介して、上記外部メモリMMRYに供給される。上記物理アドレスが外部バスEX-busのアドレスバスへ出力されることに同期して、上記中央処理装置CPUから出力される図示されないリード制御信号がシステムバスS-bus内の制御バス、キャッシュバスC-busの制御バス、バスステートコントローラBSC、入出力回路EXIF及び外部バスEX-busの制御バスを介して上記外部メモリMMRYに供給される。上記外部メモリMMRYは、上記物理アドレスによってリードアクセスされ、その物理アドレスに格納されているデータ（第1例外ハンドラの先頭の命令データ）が、外部バスEX-busのデータバス、入出力回路EXIF、バスステートコントローラBSC、キャッシュバスC-busのデータバス及びシステムバスS-bus内のデータバスを介して中央処理装置CPUへ供給されて実行される。

【0069】上記プログラムカウンタPCにセットされるベクタポイントを生成するための前記定数発生回路CVG、シフトSFT、及び算術論理演算器ALUは、中央処理装置CPUの演算部100に含まれる。

【0070】定数発生回路CVG及びシフトSFTには、割り込み通知信号SIG1、TLBミス例外の発生を示す通知信号SIG4a、及びTLBミス例外以外の一般例外の発生を示す通知信号SIG6が供給される。通知信号SIG4aは、メモリマネージメントユニットMMU内の制御部TLBCから出力される前記信号SIG4に含まれる一つの信号である。すなわち、前記信号SIG4は、複数ビットの信号を含んでおり、前述のように、メモリマネージメントユニットMMUにおけるアドレス変換に起因する例外の発生を中央処理装置CPUに通知する信号である。前記信号SIG4には、発生した例外事象がTLBミス例外であることを示す通知信号SIG4a、発生した例外事象がTLBミス例外以外の例外事象であることを示す通知信号SIG4bなどが含まれる。通知信号SIG6は、図示されるように、メモリマネージメントユニットMMUにおけるアドレス変換に際してTLBミス例外以外の例外事象が発生したとき活性化される前記信号SIG4bと、中央処理装置CPU内部において例外事象が発生したとき活性化される内部信号SIG6aとを受けるオア回路ORの出力信号とされる。

【0071】前記定数発生回路CVG及びシフトSFTにおいて、発生すべき定数とシフト量は、前記信号SI

G1, SIG4a, SIG6の内のどの信号が活性化されたかに基づいて制御される。定数発生回路CVGは、例えば、バイナリコードで2ビットの定数H'1(01)又はH'3(11)を発生する。定数発生回路CVGは、通知信号SIG4a又は通知信号SIG6が活性化されると、定数H'1を発生する。定数発生回路CVGは、割り込み発生の通知信号SIG6が活性化されると、定数H'3を発生する。シフタSFTは、通知信号SIG6が活性化されると、定数H'1(01)に対して8ビット左シフトを行うことによってベクタオフセットH'100(1.0000.0000)を出力する。TLBミス例外の通知信号SIG4aが活性化されると、シフタSFTは定数H'1(01)に対して10ビット左シフトを行うことによってベクタオフセットH'400(100.0000.0000)を出力する。通知信号SIG1が活性化されると、シフタSFTは定数H'3(11)を9ビット左シフトを行うことによって、ベクタオフセットH'600(110.0000.0000)を出力する。特に、本実施例において、ベクタオフセットとしてH'100, H'400, H'600を割り当てたのは、ベクタオフセットにおいて論理値"1"とされるビットの数を極力少なくするため、換言すれば、定数発生回路CVGの論理規模を小さくするためである。

【0072】図14から明らかなように、ハードウェア(回路)で生成すべきベクタオフセットは3種類であるから、そのハードウェア量は2ビットの定数発生回路CVGとシフタSFTだけで済む。シフタSFTはベクタオフセットの生成に専用化される必然性は全くなく、その他の演算に汎用的に利用可能にすることは容易である。

【0073】《例外処理を指標する例外要因コード》  
例外要因コード(exception code)は、図8に示されるように、夫々の例外処理の要因を識別可能にするために、夫々の例外事象に対し異なる値の例外要因コードが割り当てられる。この例外要因コードは、デバッグなどにおいて、発生した例外の要因をソフトウェア的に判定するために利用されると共に、後述されるように例外ハンドラへの分岐のためのアドレスオフセット情報としても利用される。デバッグにおいて利用可能にするため、レジスタEXPEVT及びINTEVTは、キャッシュバスC-busに結合され、バスステートコントローラBSC、入出力回路EXIF及び外部バスE-busを介して、マイクロコンピュータMPUの外部からアクセス可能にされる。すなわち、レジスタEXPEVT及びINTEVTに記憶された要因コードは、マイクロコンピュータMPUの外部に出力可能にされる。それによって、デバッグが、発生した例外の要因を要因コードから簡単に判別可能なる。

【0074】図8に示される例外要因コードは、H'20毎に割り当てられる。これをアドレス換算すれば、3

2バイト毎のアドレスオフセットとしての意味を持つ。例えば、1命令が2バイトの固定長命令であれば、16命令の記述に相当するアドレス範囲とされる。また、図8において、一般例外のうち、アドレスエラー例外(Address Error)、TLBミス例外(TLB Miss)、TLBインバリッド例外(TLBInvalid)、TLBプロテクト違反例外(TLB Protection Violation)については、ロード命令実行時(load)とストア命令実行時(store)とで別の例外要因コードが割り当てられる。これは、デバッグの容易性を考慮して設定されている。尚、図8において、内蔵周辺モジュールの割り込み要求に関する例外要因コードについては、代表的に一つの要因コードH'400だけが示されている。しかし、実際には、内蔵周辺モジュールの数およびその機能に依存して多数存在すると理解されたい。

【0075】図9に示される例外要因レジスタEXPEVTは、上記例外要因コードのうちリセット及び一般例外事象に割り当てられた例外要因コードをビット11～ビット0に格納する32ビットのレジスタである。一方、割り込み要因レジスタINTEVTは、上記例外要因コードのうち割り込み要求に割り当てられた例外要因コードをビット11～ビット0に格納する32ビットのレジスタである。これらのレジスタEXPEVT, INTEVTは、図1にも示されるように、システムバスS-busとキャッシュバスC-busとに結合され、メモリマネージメントユニットMMUの制御部TLBC内に配置されている。その理由は、TLBミスなどの高速処理が要求される一般例外の発生を一番早く検出できるのがメモリマネージメントユニットMMUの制御部TLBCだからである。

【0076】例外要因コードを格納するレジスタを一般例外と割り込み要求とで個別に設ける理由は、次の理由による。すなわち、一般例外は中央処理装置CPUの動作に同期して発生し、割り込み要求は中央処理装置CPUの動作とは非同期で発生する。また、中央処理装置CPUの動作クロック周波数は、割り込み要求の発生元とされる内蔵周辺モジュール及び外部回路の動作クロック周波数に対して複数倍、例えば、4倍のクロック周波数とされる。このように、割り込み要求と一般例外の発生タイミングの相違、および、割り込み要求発生元と例外発生元の動作速度の相違を考慮した場合、双方の例外要因コードを同じようなタイミングで同一のレジスタに書き込ませることは、レジスタに対する例外要因コードの書き込みタイミングの設定処理などを煩雑にする虞があるからである。

【0077】図15は、上記例外要因コードを生成するための論理ブロックの構成を示している。同図に示される論理回路LOGは、メモリマネージメントユニットMMUの制御回路TLBCに含まれる。この論理回路LOGには、割り込み要因A、中央処理装置CPU内部での

例外発生状況B、メモリマネージメントユニットMMU内部での例外発生状況C、リセット指示D、その他付帯情報Eが供給され、これらの入力情報の状態に応じて対応する例外要因コードを生成する。

【0078】割り込み要因Aは、割り込みコントローラINTCから信号SIG2によって論理回路LOGへ供給される。その他の付帯情報Eには前記割り込み受け付けの通知信号SIG3が含まれる。論理回路LOGは、上記割り込み要因Aとその他付帯情報Eの活性化状態に应答して、割り込み要因を対応する例外要因コードを生成し、生成した例外要因コードを割り込み要因レジスタINTEVTにセットする。

【0079】前記中央処理装置CPU内部での例外発生状況Bは、予約命令例外 (Reserved Instruction Exception)、スロット不当命令例外 (Illegal Slot InstructionException)、トラップ命令例外 (Unconditional Trap)、及び命令ブレイク例外 (User Breakpoint Trap)の発生状況を区別するための複数ビットの信号として、中央処理装置CPUから論理回路LOGに供給される。例外発生状況Bは、図1に示される制御信号SIG5に含まれるとみなされる。

【0080】メモリマネージメントユニットMMU内部での例外発生状況Cは、アドレスエラー例外 (Address Error)、TLBミス例外 (TLB Miss)、TLBインバリッド例外 (TLB Invalid)、TLBプロテクト違反例外 (TLB Protection Violation)、TLBイニシャルページライト例外 (Initial Page Write)の発生状況を区別するための複数ビットの内部信号として、論理回路LOGに供給される。

【0081】前記付帯情報Eは、アドレス変換に係る命令がロード命令かストア命令かを区別するための情報を含む。この情報は、中央処理装置CPU内で実行されるべき命令の解釈結果によって得られ、中央処理装置CPUから論理回路LOGに供給される。

【0082】論理回路LOGは、前記中央処理装置CPU内部での例外発生状況B、メモリマネージメントユニットMMU内部での例外発生状況C及び前記付帯情報Eに基づいて、発生された一般例外に対応する例外要因コードを生成し、生成された例外要因コードを例外要因レジスタEXPEVTにセットする。

【0083】リセットに関する例外要因コードは、リセット信号によって形成されたリセット指示Dに論理回路LOGが应答し、論理回路LOGが生成する。生成されたリセットに関する例外要因コードは、論理回路LOGによって例外要因レジスタEXPEVTへセットされる。リセット指示Dは、図1に示される制御信号SIG5に含まれるとみなされる。

【0084】《例外ハンドラの配置》 上記H'100、H'400、H'600で規定されるベクタオフセットは、ベクタベースレジスタVBRの値 (ベクタテーブ

ルのベースアドレス)からの固定されたオフセットを規定する。図2に示されるように、リセットに関する例外ハンドラは、H' A00000000の固定番地に配置される。TLBミス例外に関する例外ハンドラは、レジスタVBRの値+ベクタオフセット (H'400)のアドレスに直接配置される。その他の一般例外及び割り込み要求に関する例外ハンドラは、レジスタVBRの値にH'100又はH'600のベクタオフセットを加算して得られるアドレス位置に第1のハンドラ (図2に示される一般例外に共通のハンドラおよび割り込み要求に共通のハンドラ)が配置される。第1のハンドラから当該要因に固有の別の第2のハンドラへの分岐は、レジスタEXPEVTまたはINTEVTに記憶された例外要因コードをアドレスオフセットとして利用することによって行われる。

【0085】アドレスオフセットとしてどのように例外要因コードを利用するかは、ベクタベースレジスタVBR+ベクタオフセットの位置に配置されたハンドラ (ユーザ記述による)の記述内容 (ソフトウェアプログラム)によって決定される。例えば、VBR+ベクタオフセット+例外要因コードとして第2のハンドラへ分岐することを規定したり、或いは当該ハンドラ内部でセットしたベースアドレスに例外要因コードを加算して第2のハンドラへ分岐することを規定したり、又は例外例外要因コードを所定ビット数だけシフトして第2のハンドラへ分岐することを規定したりすることができる。要は、夫々の例外要因コードがH'20のような所定の間隔を空けて割り振られているので、ハンドラの記述量などに応じて当該例外要因コードを所望に利用すればよい。レジスタEXPEVTまたはINTEVTに記憶された例外要因コードをアドレスオフセットとして利用する場合、中央処理装置CPUは、システムバスS-busのアドレスバスにレジスタEXPEVTまたはINTEVTのアドレス信号を供給し、システムバスS-busの制御バスにリード制御信号を供給して、レジスタEXPEVT又はINTEVTをアクセスする。リードアクセスされたレジスタEXPEVT又はINTEVTは、その内部に記憶された要因コードをシステムバスのデータバスへ供給し、上記CPUが、その要因コードを、例えば、1つの汎用レジスタ (R0)に格納する。それによって、上記中央処理装置CPUは、要因コードを計算に使用可能になる。なお、実際の計算は、中央処理装置CPU内部の前記算術論理演算器ALUによって行われる。

【0086】例えば、図2に示されるように、TLBミス例外以外の一般例外のための例外ハンドラに関しては、当該一般例外に共通の第1のハンドラが、VBR+ベクタオフセット (H'100)のベクタポイントに配置される。当該第1のハンドラは、その例外要因に対応する例外要因コードをアドレスオフセットとしてさらに

別のハンドラ（第2のハンドラ）へ分岐することを規定する記述を有する。したがって、ベクタポイントに例外要因コードを加算して得られるアドレスのそれぞれに例外要因に固有の第2のハンドラをそれぞれ配置しておくことによって、所望の第2のハンドラへ分岐することができる。この場合、例外要因コードはレジスタEXPEVTから取得される。

【0087】また、割り込み要求に関する例外ハンドラも上記同様であり、VBR+ベクタオフセット（H' 600）のベクタポイントに、当該割り込み要求に共通の第1のハンドラが配置される。当該第1のハンドラは、その割り込み要因に対応する要因コードをアドレスオフセットとしてさらに別のハンドラ（第2のハンドラ）へ分岐することを規定する記述を有する。したがって、ベクタポイントに例外要因コードを加算して得られるアドレスのそれぞれに割り込み要因に固有の第2のハンドラをそれぞれ配置しておくことによって、所望の第2のハンドラへ分岐することができる。この場合、割り込み要求の例外要因コードは、レジスタINTEVTから取得される。

【0088】本実施例に従えば、各種例外ハンドラはTLBを利用したアドレス変換を要しない前記アドレス領域P1又はP2に割り当てられるので、アドレス変換に係る例外事象が多重に発生する事態を回避でき、これによって、データ処理の効率化に寄与することができる。

【0089】上記の様な例外ハンドラの配置において、TLBミス例外、その他の一般例外、又は割り込み要求が発生した時、中央処理装置CPUの処理は、VBRレジスタに設定されているベクタテーブルのベースアドレスにベクタオフセットをオフセットアドレスとして加算したアドレス位置に分岐され、そのアドレス位置に配置されているハンドラを処理することになる。この分岐は、マイクロコンピュータMPUのハードウェアによって規定される。そのとき、どのベクタオフセットを用いるかは、図14で説明されたように、例外発生要因の種類によって一義的に決定される。発生した例外がTLBミス例外の場合、VBR+ベクタオフセット（H' 400）で指定されるベクタポイントに割り付けられたハンドラはTLBミス例外ハンドラとされる。その他の一般例外又は割り込み要求の場合、VBR+ベクタオフセットで指定されるベクタポイントのハンドラは、その例外要因の要因コードをオフセットとして分岐する処理を含み、当該分岐先にその例外要因に固有のハンドラが配置されている。

【0090】トラップ命令を実行することによって発生するトラップ例外に関しては、そのトラップ命令が保有する8ビットのイミディエイト値が、図9に示されるトラップレジスタTRAに格納され、その値がVBRレジスタの下位のアドレスデータに加算され、加算されて得られた値によって対応する例外ハンドラがフェッチされ

るようになっている。トラップ命令例外は、例えば、ユーザープログラム内のタスクからオペレーティングシステムへのシステムコールに利用される。

【0091】《例外処理フロー》 図10は、上記ベクタオフセットによってベクタポイントまで分岐する処理のフローチャートが示される。ベクタポイントへの分岐は、図14で説明したように、マイクロコンピュータMPUのハードウェアによって規定され、特権状態への遷移として位置付けられる処理であり、以下のステップを含む。

【0092】〔例外／割り込みの検出〕 リセット、一般例外、又は割り込み要求が発生されると、それが制御部CTRLにて検出される（S1）。例えば、制御部CTRLは、マニュアルリセットであればリセットスイッチの操作によって発生するリセット信号、TLBミス例外であればメモリマネージメントユニットMMUの制御回路TLBCから出力されるTLBミスの検出信号など、割り込み要求であれば割り込みコントローラINTCから中央処理装置CPUに供給される割り込み信号SIG1に基づいて例外を検出する。中央処理装置CPU内部の例外は、その内部状態に基づいて検出される。

【0093】〔PC、SRの退避〕 次に、中央処理装置CPUは、プログラムカウンタPCの値をセーブプログラムカウンタSPCに退避し、ステータスレジスタSRの値をセーブステータスレジスタSSRに退避する（S2）。退避のタイミングは、一般例外の場合、当該例外が発生した時点とされ、したがって、現在実行完了されていない命令のアドレスなどが退避され、復帰後はその命令が再実行されることになる。割り込み要求の場合の退避のタイミングは、その割り込み要求が発生した時点において、中央処理装置CPUがその時点に実行していた命令の実行完了を待って上記2つの値の退避が行われ、その結果として、その命令の次の命令のアドレスなどが退避されることになる。これにより、例外が発生したときプログラムカウンタPCの値やステータスレジスタSRの値をメモリ上のスタック領域に退避するためのメモリアクセスを要せず内蔵レジスタアクセスで済むため、本発明のマイクロコンピュータMPUの退避処理を高速に行うことができる。

【0094】〔プロセッサモードの切換え〕 中央処理装置CPUは、ステータスレジスタSRのモードビットMDを論理値1にしてマイクロコンピュータMPUの動作モードを特権モードに設定する（S3）。中央処理装置CPUは、更に、ステータスレジスタSRのブロックビットBLを論理値“1”にセットして、その他の例外及び割り込みをマスクする（S3）。中央処理装置CPUは、さらに、ステータスレジスタSRのレジスタバンクビットRBを論理値1にセットして、特権状態における汎用レジスタR0～R7をユーザ状態とは別のレジスタバンクに切り換える（S3）。したがって、汎用レジ

スタR0～R7のバンク切換えを行うから、ユーザ状態におけるレジスタバンクの汎用レジスタR0～R7を回避する処理は不要になる。

【0095】〔例外要因のセット〕 次に、そのときの例外や割り込み要求の要因に割り当てられている例外要因コードが、図15において説明されたように、レジスタEXPEVT又はINTEVTにセットされる（S4）。前者のレジスタEXPEVTには一般例外要因又はリセットの要因コードが、後者レジスタINTEVTには割り込み要求の要因コードが、MMU内の論理回路LOGによってセットされる。

【0096】〔ベクタポイントに分岐〕 上述のように、中央処理装置CPUの制御は、所定のベクタポイントに分岐される（S5）。例外処理がリセットである場合、固定番地H'A00000000に分岐される。一般例外や割り込み要求の場合、レジスタVBRのベクタテーブルベースアドレスに所定のベクタオフセットが加算されたアドレスに、中央処理装置CPUの制御が分岐される。分岐されたベクタポイントには、図2に示されるように、所定のハンドラが置かれている。リセットの場合、リセットのための例外ハンドラが配置され、TLBミス例外の場合、TLBエントリ更新などのためのTLBミス例外ハンドラが置かれている（図2参照）。TLBミス例外以外の一般例外の場合、あるいは、割り込み要求の場合、例外要因に固有の別のハンドラに分岐するための処理を含むハンドラ（図2に示される一般例外に共通なハンドラ、割り込み要求に共通なハンドラ）が配置されている。

【0097】命令実行中にTLBミス例外が発生したときには、TLBミスに係る論理ページアドレスに対応する物理ページアドレスを外部メモリMMRY上のページテーブルから読み出してアドレス変換バッファTLBを更新し、物理アドレスへの変換を行わなくてはならないため、そのTLBミス例外の発生からその対処への遷移時間は命令実行速度若しくは中央処理装置CPUによるデータ処理速度に大きな影響を与える。したがって、データ処理性能にもっとも大きな影響を与えられとされるTLBミス例外ハンドラを上記ベクタポイントに直接配置することは、中央処理装置CPUの制御を例外ハンドラに引き渡すまでの処理時間を著しく短縮させる。それによって、マイクロコンピュータMPUのデータ処理速度の高速化が実現できる。

【0098】図11は、上記ベクタポイントに置かれた例外ハンドラ（リセットハンドラやTLBミス例外ハンドラを除く）の処理フローの一例を示している。この処理は、上述のように、専らソフトウェアによって規定され、以下の処理ステップを含む。

【0099】まず、後から発生した別の例外や割り込み要求を多重に受け付けるか否かが決定される（S11）。その決定は、当該ハンドラの記述に従う。

【0100】多重に割り込み要求などを受け付けない場合、レジスタEXPEVT、INTEVTから例外要因コードをリードし（S12）、例外要因コードをアドレスオフセットとして利用して、対応する例外ハンドラに分岐するための処理が行われる（S13）。中央処理装置CPUが、VBR+ベクタオフセット+例外要因コードの計算を実行し、その計算されたアドレスに割付けられる例外ハンドラ、すなわち、分岐先の例外ハンドラへ処理が移行される。それによって中央処理装置CPUが、発生した例外に対処する。

【0101】多重に割り込み要求などを受け付ける場合、図10で説明した処理フローを保証するために、中央処理装置CPUは、セーブステータスレジスタSSR、セーブプログラムカウンタSPCなど、図10のハンドラで利用するレジスタのデータを外部メモリMMRYに退避する（S14）。中央処理装置CPUは、更に、ステータスレジスタSRのブロックビットBLを論理値“0”にセットして、その他の例外及び割り込みに対するマスクを解除する（S15）。その後、上記同様、中央処理装置CPUは、レジスタEXPEVT、INTEVTから例外要因コードをリードし（S16）、例外要因コードをアドレスオフセットとして対応する例外ハンドラに分岐される（S17）。すなわち、中央処理装置CPUが、VBR+ベクタオフセット+例外要因コードの計算を実行し、その計算されたアドレスに割付けられる例外ハンドラ、すなわち、分岐先の例外ハンドラへ処理が移行される。それによって中央処理装置CPUが、発生した例外に対処する。

【0102】図10及び図11のフローチャートから明らかなように、TLBミス例外に対するように高速処理を最優先とするものに対しては、レジスタVBRの値を1回リードしてそれにベクタオフセット（H'400）を加算するだけでTLBミス例外ハンドラのベクタポイントを取得できる。また、別の例外要因に対しては、レジスタVBRの値にベクタオフセットを加算して得られるベクタポイントの取得に加えて、レジスタEXPEVT又はINTEVTの値を1回リードして得た例外要因コードをアドレスオフセットとする事により所要の例外ハンドラに分岐できる。この分岐においても、分岐先アドレスを直接メモリアクセスによって取得することなく高速に例外に応ずる処理に制御を移すことができる。

【0103】図11からも明らかなように、多重に例外や割り込みを受け付けるか否かはハードウェアによって規定されたベクタポイントのアドレスに割り当てられたハンドラの記述によって決定される。例外の多重受け付けを許容する場合には、退避レジスタSSR、SPCに既に退避されている値を外部メモリMMRYへ退避する。TLBミス例外の様に高速処理が必要な場合、多重の例外や多重割り込みを禁止したまま、TLBミス例外ハンドラを実行するので、セーブステータスレジスタS



S R、セーブプログラムカウンタ S P C のメモリへの退避を一切要せず、この点において、T L B ミス例外の発生から T L B ミスハンドラへの遷移時間を最短にする事を保証している。

【0104】図12は、一般例外や割り込み要求に対処するための例外処理へ分岐した後、分岐先において、例外や割り込みが新たに発生した場合の処理フローを示している。

【0105】例外や割り込みが新たに発生したとき (S 2 1)、ステータスレジスタ S R のブロックビット B L が論理値“1”であるか否か (S 2 2)、そして当該新たに発生した例外や割り込みが命令ブレイク例外であるか否か (S 2 3) によって、当該新たに発生した例外や割り込みを受け付ける否かが決定される。S R、B L = 0 によって多重受け付けが許容されている場合には図10、図11のフローに従って処理される (S 2 4)。

【0106】上記 S R、B L = 1 によって例外の多重受け付けが禁止されている場合 (S 2 2 の Y E S)、そのときに発生した一般例外が命令ブレイク例外でなく (S 2 3 の N O)、且つ、それが割り込み要求の場合 (S 2 5 の Y E S)、ソフトウェアが S R、B L = 0 にするまで (例えば、前記ステップ S 1 4 の処理を終えてステップ S 1 5 に進むまで)、当該割り込みの受け付けが抑止される (S 2 6)。それが一般例外やリセットの場合 (S 2 5 の N O)、例外要因コードをレジスタ E X P E V T にセットした後、自動的にリセットハンドラに分岐される (S 2 7)。

【0107】そのときの例外事象が命令ブレイク例外である場合、図10のハードウェア処理において、S P C、S S R、E X P E V T の各レジスタは更新されず、中央処理装置 C P U の処理は命令ブレイクハンドラに分岐し、当該ハンドラの処理によってレジスタ S P C、S S R をメモリに退避する。そして、さらに、ブレイクレジスタ I B R のブレイクポイントアドレスから復帰アドレスが計算され、当該復帰アドレスがセーブプログラムカウンタ S P C にセットされる。そして、命令ブレイク例外に対処するための命令ブレイク例外ハンドラの処理が行われる。命令ブレイク例外ハンドラからの復帰は、レジスタ S P C に退避された復帰アドレスが利用され、命令ブレイクポイントの次からデバッグ対象プログラムを継続して実行することができる。このように、命令ブレイク例外に対しては、多重に発生する例外の受け付けが禁止されている状態のデバッグ対象プログラムに対しても任意の位置で命令ブレイクをかけてシステム評価やプログラムデバッグを行うことができる。

【0108】上記実施例によれば以下の作用効果がある。〔1〕 T L B ミス例外に対するように高速処理を最優先とするものに対しては、レジスタ V B R の値を1回リードし、その値にベクタオフセット (H' 4 0 0) を加算するだけで T L B ミス例外ハンドラのベクタポイ

ント (分岐先アドレス) を取得できる。また、別の例外要因に対しては、レジスタ V B R の値にベクタオフセットを加算して得られたベクタポイントに、さらに、レジスタ E X P E V T 又は I N T E V T の値を1回リードして得た例外要因コードをアドレスオフセットとして加算する事により、所要の例外ハンドラの先頭アドレス (分岐先アドレス) を取得できる。このように、分岐先アドレスを外部メモリのアクセスによって取得しないため、例外発生からその例外に対処するためのハンドラの処理へ遷移するまでの遷移時間が短縮される。これによって、マイクロコンピュータ M P U の全体的なデータ処理速度の高速化が実現できる。換言すれば、データ処理の高速化に密接な T L B ミス例外のような処理に対しては、その例外処理への遷移時間を短縮することが最優先とされ、例外処理への遷移時間の短縮が T L B ミス例外データ処理速度の高速化に寄与しないと考えられる割り込みなどに対しては、ハンドラのマッピングやハンドラのサイズに対して融通性を持たせ、ユーザの使い勝手の向上が計られる。

【0109】〔2〕ベクタベースレジスタの値とベクタオフセットとを加算して得られたアドレスに配置された例外ハンドラの内の一部の例外ハンドラ (T L B ミス例外ハンドラ) だけが、対応する例外に対して直接的な例外処理を行なう。その他の例外に対しては、例外要因コードをアドレスオフセットに利用して、更に、分岐されて所定のハンドラにたどり着くようになっている。換言すれば、ベクタベースアドレスに対する固定のオフセットであるベクタオフセットの数は例外要因の数よりも遥かに少ない。全ての例外要因に対して固有の固定オフセット (ベクタオフセット) を利用する場合、例外要因が増えるほどベクタオフセットを生成するためのハードウェア量が増大する。それゆえ、マイクロコンピュータの形成される半導体チップの面積は増加し、マイクロコンピュータ自体のコストも、高くなってしまふと考えられる。しかしながら、本発明のような例外処理においては、物理的な回路規模の縮小とデータ処理の高速化との双方を満足させることができる。したがって、本発明のマイクロコンピュータ M P U は、小面積の半導体チップに形成することができ、そのコストも低減される。

【0110】〔3〕多重に例外や割り込みを受け付けるか否かは、ハードウェアによって規定されたベクタポイントのハンドラの記述によって決定される。多重な例外の受け付けを許容する場合、退避レジスタ S S R、S P C に既に退避されている値が、さらに、外部メモリ M M R Y に退避される。T L B ミス例外の様に高速処理が必要な場合、多重例外や多重割り込みを抑止した状態で、例外処理ハンドラが実行されるので、セーブステータスレジスタ S S R、セーブプログラムカウンタ S P C の外部メモリ M M R Y への退避を一切必要としない。この点において、T L B ミス例外の発生から T L B ミスハンド

ラへの遷移時間を最短にする事が保証される。

【0111】〔4〕ハードウェアで決定されるベクタポイントのハンドラから、更に、別のハンドラに分岐する際、例外要因コードが分岐のためのアドレスオフセットとして利用される。換言すれば、夫々の例外要因コードをアドレスオフセットとして利用できるように、例えば、アドレス換算で32バイト分に相当するようなH'20の間隔を置いて例外要因コードが割り当てられている。そのため、例外要因コード（アドレスオフセット）に対するベースアドレスは、ハードウェアによって決定されているベクタポイントに配置される例外処理ハンドラのプログラム記述に従って自由に決定できることになる。このことは、実際の分岐先アドレスや分岐先のハンドラのサイズ対してユーザの自由度を保証する。

【0112】〔5〕内部状態及び復帰命令アドレスを退避するレジスタSSR、SPCを採用することにより、例外発生時の退避に際して外部メモリへアクセスを減らすことができる。

【0113】〔6〕中央処理装置CPUの動作に同期して発生する一般例外と非同期で発生する割り込み要求との双方に対処するため、一般例外に関する例外要因コードを格納するレジスタEXPEVTと割り込み要求に関する例外要因コードを格納するレジスタINTEVTとが設けられる。双方の例外発生に対して同じようなタイミングで要因コードを共通のレジスタに書き込むための煩雑な処理を避けられ、例外要因コードを各レジスタにセットする処理を簡素化できる。

【0114】〔7〕リセット例外ハンドラやTLBミス例外処理ハンドラ（例外要因に応じてハードウェアで決定されている分岐先のハンドラ）の処理において、初期的なプロセッサモードが、例外の多重受け付け抑止状態及び特権状態への移行によって一定にされる。したがって、ユーザの当該ハンドラ上で規定できる例外処理の内容は高い自由度を保証される。

【0115】〔8〕汎用レジスタR0～R7のレジスタバンクを特権状態においてのみソフトウェアで切り換え可能にすることにより、中央処理装置CPUは、特権状態において、ユーザ状態とは異なるバンクの汎用レジスタR0～R7を利用できる。このことは、例えば、例外処理においてユーザ状態から特権状態にプロセッサモードを切り換える時、汎用レジスタR0～R7の内容の外部メモリMMRYへの退避を省略できることを意味する。この点においても、例外発生からそれに対処する処理へ遷移するまでの遷移時間が短縮される。

【0116】〔9〕多重例外を抑止した状態の例外ハンドラの処理において、命令ブレイク例外を処理可能にすることにより、多重に発生する例外の受け付けが禁止されている状態のデバッグ対象プログラムに対しても任意の位置で命令ブレイクをかけてシステム評価やプログラムデバッグを行うことができるを保証できる。

【0117】〔10〕アドレス領域P1、P2は、対応する物理アドレスが固定とされる領域であり、アドレス変換バッファTLBを利用したアドレス変換の対象とはされない。特権状態において、アドレス領域P1、P2をアクセスするときには、TLBミスを初めとするアドレス変換に係る例外を多重に生ずることはない。したがって、割り込みや一般例外などの各種例外ハンドラをアドレス領域P1又はP2に配置することにより、TLBミスを初めとするアドレス変換に係る例外が例外処理中に新たに生ずることがなくなる。換言すれば、アドレス変換に係る例外が多重に発生する事態を回避できるので、中央処理装置CPUのデータ処理を効率化することができる。

【0118】以上本発明者によってなされた発明を実施例に基づいて具体的に説明したが、本発明はそれに限定されるものではなく、その要旨を逸脱しない範囲において種々変更可能であることは言うまでもない。

【0119】例えば、ハードウェアで決定されるベクタオフセットはH'100、H'400、H'600に限定されず、また、アドレスオフセットを規定する要因コードやその割り当て間隔も上記実施例に限定されず、適宜変更可能である。また、例外は、上記実施例のリセット、一般例外、割り込み要求に大別する考え方に限定されない。さらに、ハードウェアによって決定された第1のベクタポイントに直接TLBミス例外ハンドラを配置したが、当該ベクタポイントに配置できる例外ハンドラは、それに限定されず、システム構成に応じて適宜変更できる。また、上記第1のベクタポイントから分岐される第2のポイントに配置した別の例外ハンドラにおいては、更に、別のハンドラに分岐できることは言うまでもない。その場合にも要因コードをアドレスオフセット情報として利用するが可能である。

【0120】以上の説明では主として本発明者によってなされた発明をその背景となった利用分野であるマイクロコンピュータに適用した場合について説明したが、データプロセッサ、マイクロプロセッサ、ディジタルシグナルプロセッサなどその称呼の如何に関わらず例外処理をサポートする各種データ処理装置に広く適用することができる。

【0121】

【発明の効果】本願において開示される発明のうち代表的なものによって得られる効果を簡単に説明すれば下記の通りである。

【0122】すなわち、ハードウェアによって決定された第1のベクタポイントのハンドラへの分岐、更に当該ハンドラから要因コードをオフセットとする別のハンドラへの分岐の何れにおいても、分岐先アドレスを直接メモリのアクセスによって取得することを要しないため、例外発生からそれに対処するハンドラへの遷移するまでの遷移時間を短縮でき、これによって、マイクロコンピ

ータ等のデータ処理装置の全体的なデータ処理速度の高速化が実現できる。

【0123】データ処理の高速化に密接なTLBミス例外のような処理に対しては、固定オフセットとしてのベクタオフセットを利用して得られたアドレスに配置された例外ハンドラによって直接例外処理を行うようにする。したがって、TLBミス例外に対する処理への遷移時間を短縮できる。一方、例外処理への遷移時間の短縮がTLBミス例外程データ処理速度の高速化に寄与しないと考えられる割り込みなどに対しては、ハンドラのマッピングやハンドラのサイズに対する融通性という点で使い勝手の向上を優先させることができる。

【0124】ベクタベースレジスタの値とベクタオフセットとを加算して得られたアドレスに配置された例外ハンドラの内の一部の例外ハンドラ（TLBミス例外ハンドラ）だけが、対応する例外に対して直接的な例外処理を行なう。その他の例外に対しては、例外要因コードをアドレスオフセットに利用して、更に、分岐されて所定のハンドラにたどり着くようになっている。したがって、物理的な回路規模の縮小とデータ処理の高速化との双方を満足させることができる。

【0125】ハードウェアによって規定されるベクタポイントのハンドラ上では、新たに発生する例外の多重受け付けを抑止するか否かを決定し、多重に受け付けるときは退避レジスタの内容をメモリに退避させることにより、例外発生時における状態退避と復帰アドレス退避を必ず退避レジスタにでき、この点においても例外発生からそれに対処するためのハンドラへの遷移時間を短縮できる。

【0126】ハードウェアで決定されるベクタポイントのハンドラから更に別のハンドラに分岐する際に要因コードが分岐のためのアドレスオフセットとして利用されるようになっているので、当該オフセットに対するベースアドレスはハードウェアによって決定されているベクタポイントのハンドラの記述に従って自由に決定でき、実際の分岐先アドレスや分岐先のハンドラのサイズに対して自由度を保證することができる。

【0127】内部状態及び復帰命令アドレスを退避するレジスタを採用することにより例外発生時の退避に際してメモリアクセスを減らすことができる。

【0128】中央処理装置の動作に対して同期して発生される第1の例外と非同期で発生される第2の例外との双方に対処するとき、中央処理装置が割り込み要求の発生元とされる周辺モジュールの動作クロック周波数に対して複数倍のクロック周波数で動作されるような場合に、例外要因コードを格納するレジスタを上記第1と第2の例外に対して格別に設けることにより、双方の例外発生に対して同じようなタイミングで要因コードを共通のレジスタに書き込むための煩雑な処理を避けられ、これによって、要因コードをレジスタにセットする処理を

簡素化できる。

【0129】例外要因に応じてハードウェアで決定されている分岐先のハンドラ上での初期的なプロセッサモードを、例外の多重受け付け抑止状態及び特権状態を以て一定にすることは、ユーザが当該ハンドラ上で規定できる例外処理の内容に高い自由度を保證できる。

【0130】汎用レジスタのレジスタバンクを特権状態においてのみソフトウェアで切り換え可能にすることにより、特権状態においてユーザ状態とは異なるバンクの汎用レジスタを利用できるようになり、例えば例外処理においてユーザ状態から特権状態にプロセッサモードを切り換えるときに汎用レジスタの内容をメモリに退避しなくても済むようにでき、この点においても例外発生からそれに対処する処理への遷移時間を短縮することができる。

【0131】例外を抑止した状態の例外ハンドラ中で命令ブレーク例外を処理可能にすることにより、多重に発生する例外の受け付けが禁止されている状態のデバッグ対象プログラムに対しても任意の位置で命令ブレークをかけてシステム評価やプログラムデバッグを行うことを保證できる。

【図面の簡単な説明】

【図1】本発明の一実施例に係るシングルチップマイクロコンピュータのブロック図である。

【図2】例外発生時におけるハードウェアによるベクタポイントとポイントされたハンドラから更に別のハンドラに分岐する本実施例の手法を概念的に示した説明図である。

【図3】バンク構成の汎用レジスタ及びシステムレジスタの説明図である。

【図4】ユーザモードにおける図3の汎用レジスタ及びシステムレジスタの状態説明図である。

【図5】特権モードにおける図3の汎用レジスタ及びシステムレジスタの状態説明図である。

【図6】コントロールレジスタの説明図である。

【図7】割り込み要因とベクタオフセットの割り当てなどを示す説明図である。

【図8】割り込み要因とそれに割り当てられた割り込み要因コードとの対応説明図である。

【図9】例外要因レジスタ及びトラップレジスタの説明図である。

【図10】ハードウェアによって決定されたベクタポイント処理の一例フローチャートである。

【図11】ベクタポイントに置かれたハンドラの処理の一例フローチャートである。

【図12】割り込み又は一般例外ハンドラによる処理において別の割り込み又は一般例外が発生した場合における処理の一例フローチャートである。

【図13】本実施例に係るマイクロコンピュータの一例アドレスマップである。

【図14】ベクタオフセットを生成するハードウェアの一例ブロック図である。

【図15】例外要因コードを生成する論理の一例ブロック図である。

【符号の説明】

MPU マイクロコンピュータ

CPU 中央処理装置

CTRL 制御部

PC プログラムカウンタ

SPC セーブプログラムカウンタ

SR ステータスレジスタ

MD モードビット

RB レジスタバンクビット

BL ブロックビット

SSR セーブステータスレジスタ

VBR ベクタベースレジスタ

R0~R15 汎用レジスタ

UBC 命令ブレイクコントローラ

IBR 命令ブレイクアドレスレジスタ

MMU メモリマネージメントユニット

TLB アドレス変換バッファ

TLBC 制御部

EXPEVT 一般例外用の例外要因レジスタ

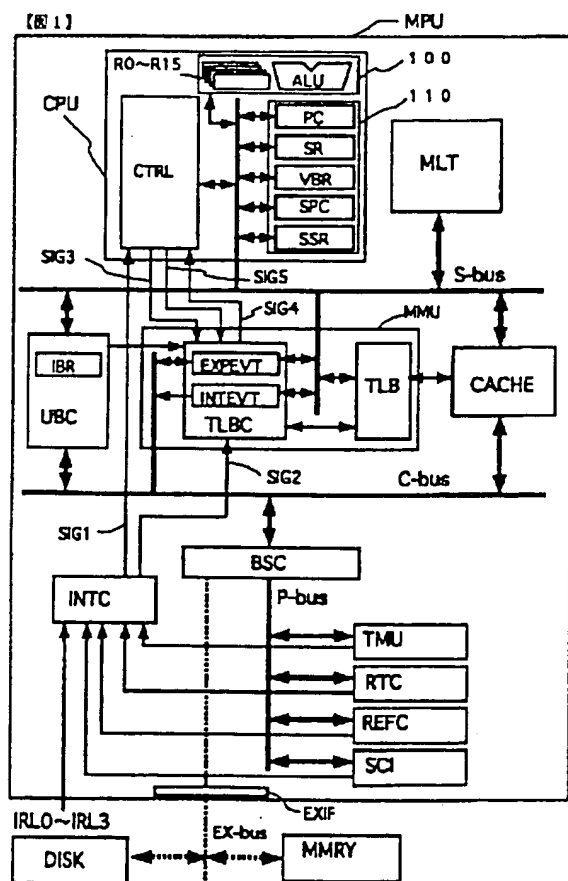
INTEVT 割り込み要求用の例外要因レジスタ

TRAP トラップレジスタ

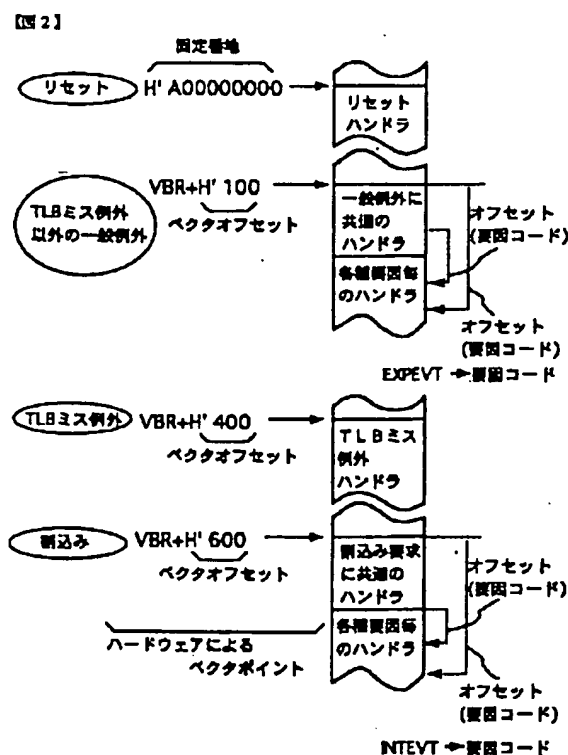
CVG 定数発生回路

SFT シフタ

【図1】

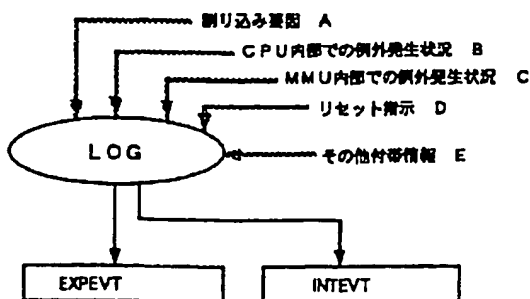


【図2】

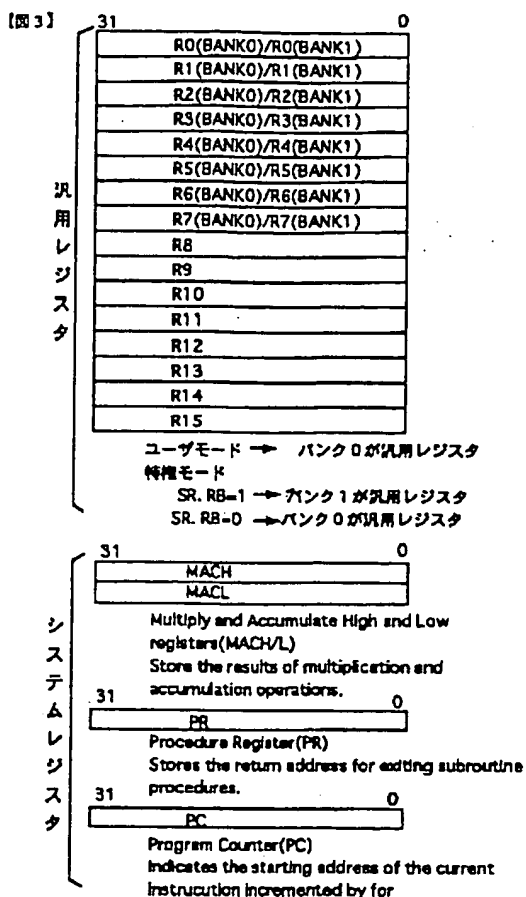


【図15】

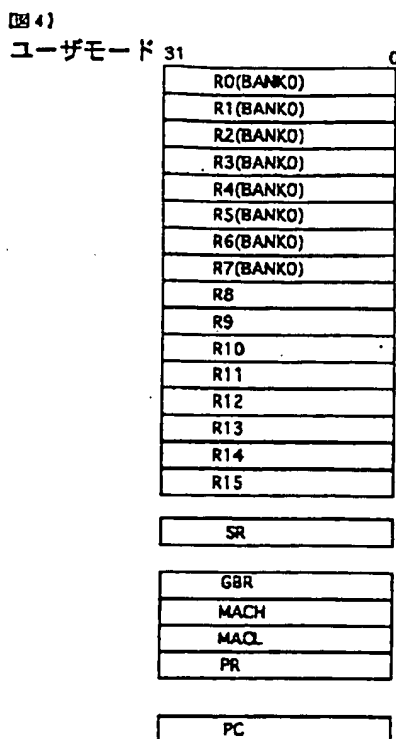
【図15】



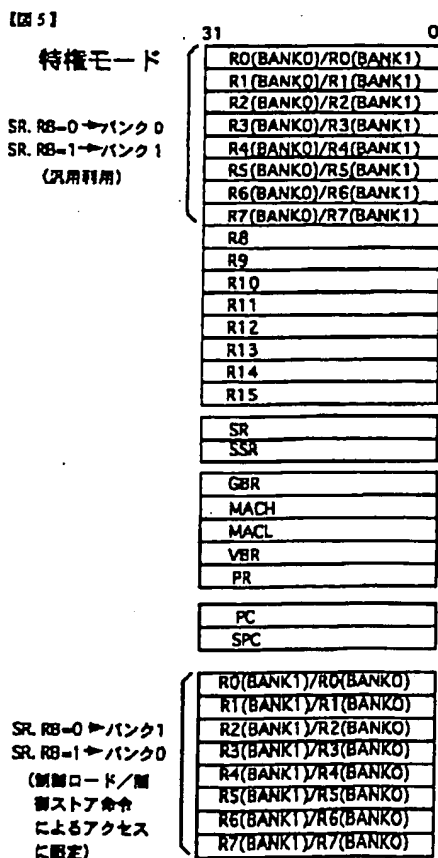
【図3】



【図4】



【図5】

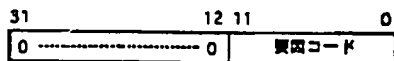


【図9】

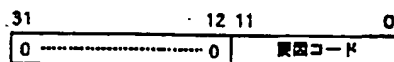
【図9】

例外原因レジスタ

EXPEVT

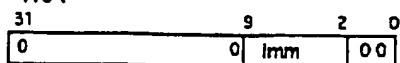


INTEVT



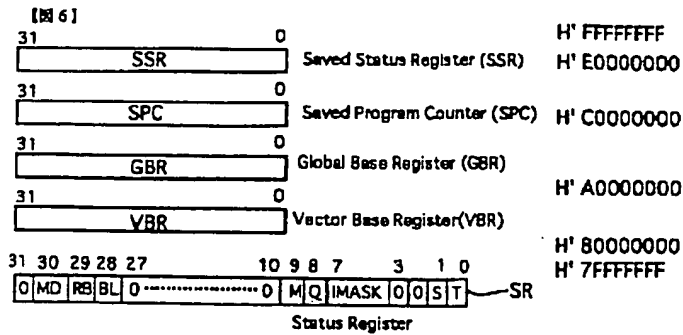
トラップレジスタ

TRA



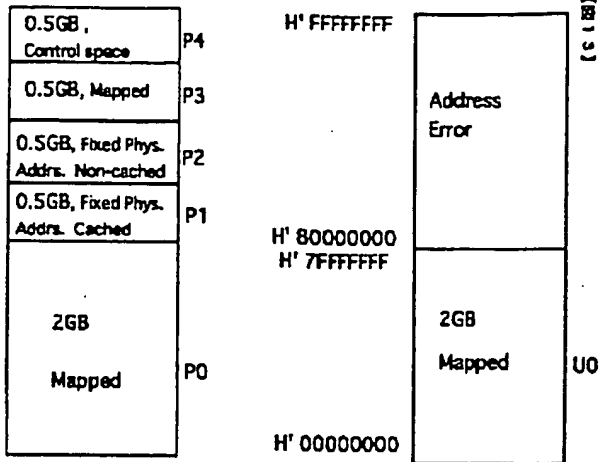
0 : reserved bits, always read as zero  
Imm : 8-bit immediate data in TRAP instruction

【図6】



T bit : Indicating a carry, borrow, overflow, or underflow.  
 S bit : Used by the MAC instruction.  
 IMASK : 4-bit field indicating the interrupt request mask level.  
 M & Q bits : Used by the DIV0 U/S and DIV1 instructions.  
 RB : Register bank bit, used to define the general-purpose registers in privileged mode. A logic-one designates R0~R7(BANK1) and R8~R15 are accessed as general-purpose register, and R0~R7(BANK0) are only accessed by LDC/STC instructions; and A logic zero designates R0~R7(BANK0) and R8~R15 are accessed as general-purpose register, and R0~R7(BANK1) are only accessed by LDC/STC instructions.  
 BL : Block bit, used to mask exceptions in privileged mode, as follows;  
 BL=1, exceptions are masked (not accepted)  
 BL=0, exceptions are accepted  
 MD : Processor Operation Mode bit, indicates the processor operation mode, as follows;  
 1= Privileged mode  
 0= User mode

H' 00000000  
 特権モード



ユーザモード  
 Mapped: TLBによるアドレス変換対象

【図7】

# Vectored Exception Events

Exception Type	Current Instruction	Exception Events	Priority Level *1	Execution Order	Vector Address	Vector Offset
Reset	Aborted	Power-On	1	-	H' A0000000	-
		Manual Reset	1	-	H' A0000000	-
General Exception Events	Aborted and Retried	Address Error(Instr. Access)	3	2	-	H' 00000100
		TLB Miss (Instr. Access)	3	3	-	H' 00000400
		TLB Protection Violation(Instr. Access)	3	5	-	H' 00000100
		Reserved Instruction Exception	3	6	-	H' 00000100
		Illegal Slot Instruction Exception	3	6	-	H' 00000100
		Address Error(Data Access)	3	7	-	H' 00000100
		TLB Miss (Data Access)	3	8	-	H' 00000400
		TLB Invalid (Data Access)	3	9	-	H' 00000100
		TLB Protection Violation(Data Access)	3	10	-	H' 00000100
		Initial Page Write	3	11	-	H' 00000100
	Completed	Unconditional Trap (TRAP instruction)	3	6	-	H' 00000100
		User Breakpoint Trap	3	n *2	-	H' 00000100
General Interrupt Request	Completed	Nonmaskable Interrupt	2	-	-	H' 00000600
		External Hardware Interrupt	4 *3	-	-	H' 00000600
		Peripheral Module Interrupt	4 *3	-	-	H' 00000600

\*1 Priority level is assigned from 1 to 4 in highest to lowest order, with level 1 the highest priority and level 4 the lowest.

\*2 A breakpoint trap may be user-defined for a preexecution instruction breakpoint (n=1), postexecution instruction breakpoint (n=12), or operand breakpoint (n=12).

\*3 Relative priority between external hardware interrupts and peripheral module interrupt is defined by software.

【図 8】

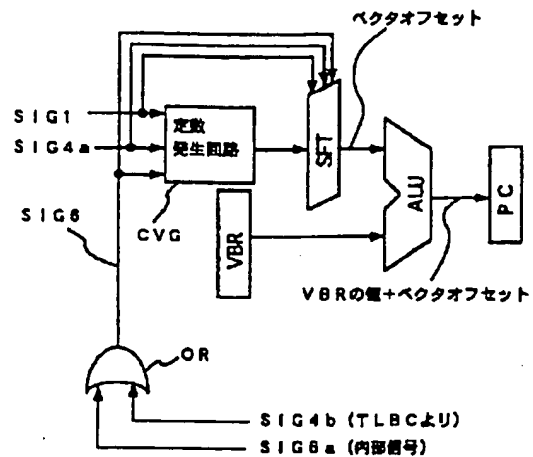
【図 8】

EXPEVT Register and INTEVT Register Exception Code

Exception Type	Exception Event	Exception Code
Reset	Power-On	H'000
	Manual Reset	H'020
General Exception Events	TLB Miss/Invalid (load)	H'040
	TLB Miss /Invalid(store)	H'060
	Initial Page Write	H'080
	TLB Protection Violation(load)	H'0A0
	TLB Protection Violation(store)	H'0C0
	Address Error(load)	H'0E0
	Address Error(store)	H'100
	Unconditional Trap (TRAP instruction)	H'160
	Reserved Instruction Exception	H'180
	Illegal Slot Instruction Exception	H'1A0
	User Breakpoint Trap	H'1E0
General Interrupt Events	Nonmaskable Interrupt	H'1C0
	External Hardware Interrupt	
	IRL3~IRLO=0000	H'200
	IRL3~IRLO=0001	H'220
	IRL3~IRLO=0010	H'240
	IRL3~IRLO=0011	H'260
	IRL3~IRLO=0100	H'280
	IRL3~IRLO=0101	H'2A0
	IRL3~IRLO=0110	H'2C0
	IRL3~IRLO=0111	H'2E0
	IRL3~IRLO=1000	H'300
	IRL3~IRLO=1001	H'320
	IRL3~IRLO=1010	H'340
	IRL3~IRLO=1011	H'360
	IRL3~IRLO=1100	H'380
	IRL3~IRLO=1101	H'3A0
	IRL3~IRLO=1110	H'3C0
	IRL3~IRLO=1111	H'3E0
	Peripheral Module Interrupt	H'400

【図 14】

【図 14】



SIG1 : 割り込み信号

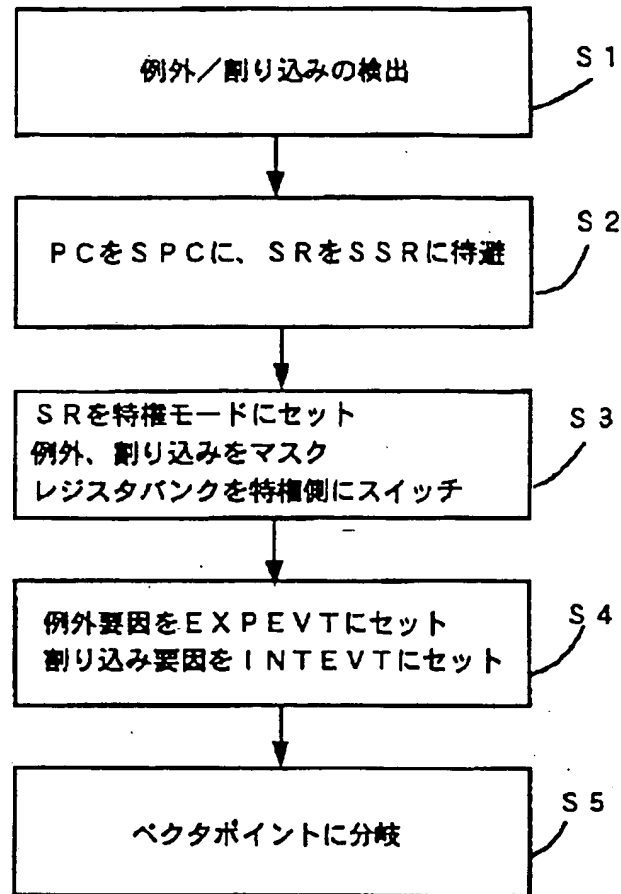
SIG4a : TLBミス例外の発生通知信号

SIG6 : TLBミス例外以外の一般例外の発生通知信号

【図10】

例外処理フロー

(1) ハードウェア (特権状態への遷移)



PC : Program Counter

SR : Status Register

SPC: Saved PC

SSR: Saved SR

EXPEVT : Exception Event Register

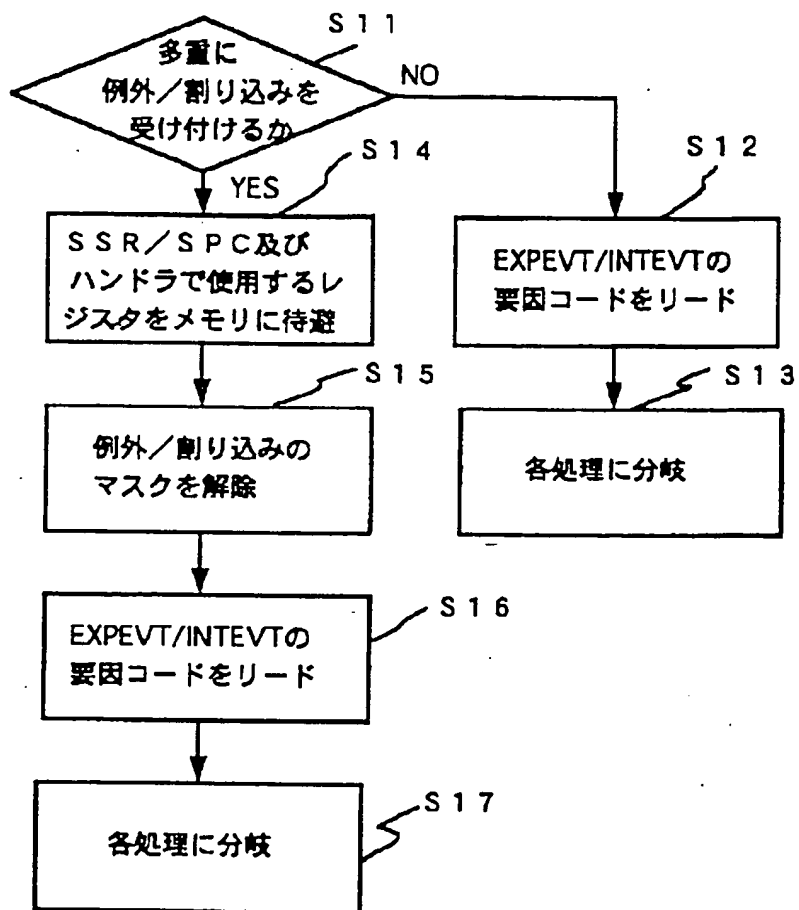
INTEVT : Interrupt Event Register



【図11】

例外処理フロー

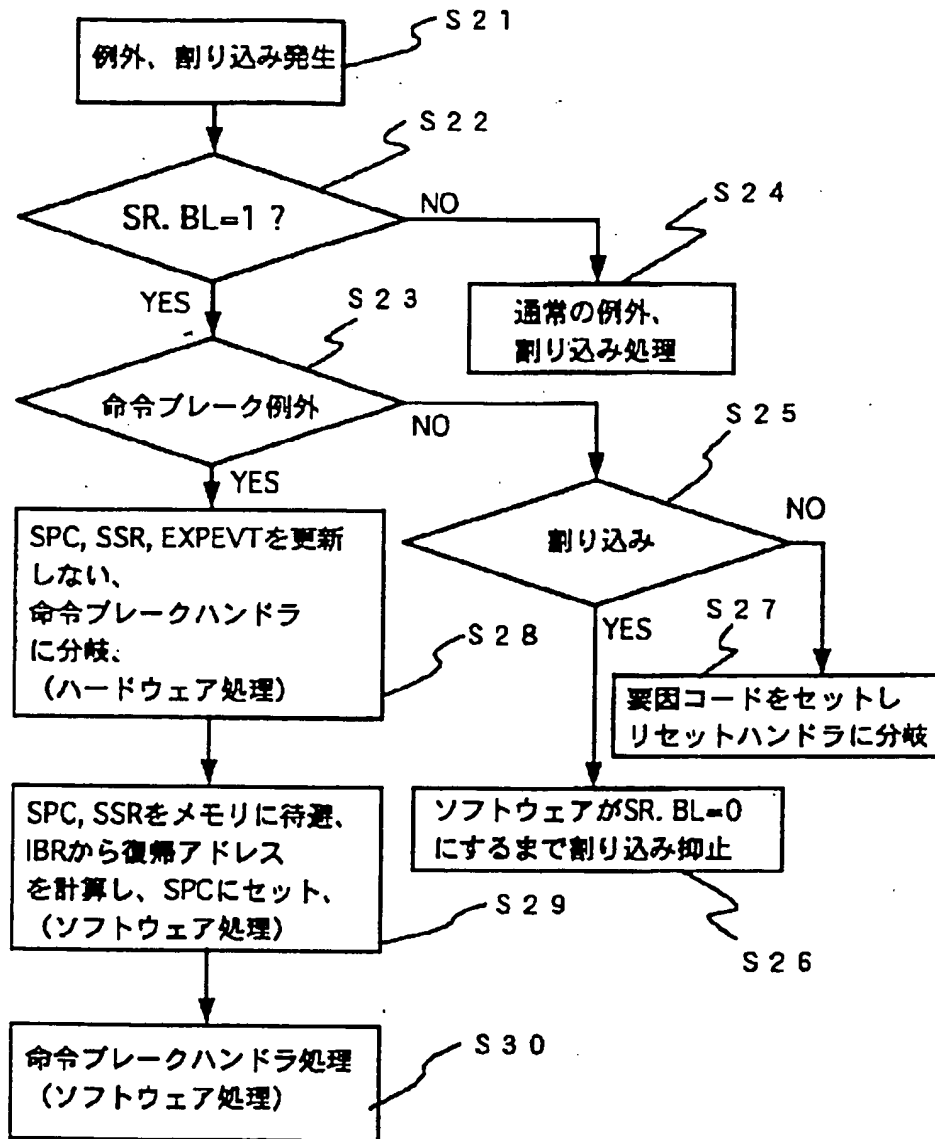
(2) ソフトウェア（ベクタポイントに置かれたハンドラの処理）



【図12】

例外処理フロー

(3) SR. BL=1のときの処理フロー



フロントページの続き

(72)発明者 成田 進

東京都小平市上水本町5丁目20番1号 株  
式会社日立製作所半導体事業部内